AD-A256 360

(12)

US Army Corps
Engineers

Hydrologic Engineering Center

Missouri River Reservoir System Analysis Model: Phase II

January 1992

92 1 . . . 054

407989
DEFENSE TECHNICAL INFORMATION CENTER
9227104
356 pgs

Approved for Public Release. Distribution Unlimited.

PR-17

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1a. REPORT SECURITY CLASSIFICATION UNCLASSIFIED	1b. RESTRICTIVE MARKINGS
2a. SECURITY CLASSIFICATION AUTHORITY	3. DISTRIBUTION/AVAILABILITY OF REPORT Distribution of this document is unlimited.
2b. DECLASSIFICATION/DOWNGRADING SCHEDULE	
4. PERFORMING ORGANIZATION REPORT NUMBER(S) PR-17	5. MONITORING ORGANIZATION REPORT NUMBER(S)
6a. NAME OF PERFORMING ORGANIZATION Hydrologic Engineering Center 6b. OFFICE SYMBOL *(If applicable)* CEWRC-HEC	7a. NAME OF MONITORING ORGANIZATION
6c. ADDRESS *(City, State, and ZIP Code)* 609 Second Street Davis, CA 95616-4687	7b. ADDRESS *(City, State, and ZIP Code)*
8a. NAME OF FUNDING/SPONSORING ORGANIZATION 8b. OFFICE SYMBOL *(If applicable)*	9. PROCUREMENT INSTRUMENT IDENTIFICATION NUMBER
8c. ADDRESS *(City, State, and ZIP Code)*	10. SOURCE OF FUNDING NUMBERS PROGRAM ELEMENT NO. / PROJECT NO. / TASK NO. / WORK UNIT ACCESSION NO.

11. TITLE *(Include Security Classification)*
Missouri River Reservoir System Analysis Model - Phase II

12. PERSONAL AUTHOR(S)
CEWRC-HEC

13a. TYPE OF REPORT	13b. TIME COVERED	14. DATE OF REPORT *(Year, Month, Day)*	15. PAGE COUNT
Final	FROM ________ TO ______	January 1992	250

16. SUPPLEMENTARY NOTATION

17. COSATI CODES		
FIELD	GROUP	SUB-GROUP

18. SUBJECT TERMS *(Continue on reverse if necessary and identify by block number)*
Prescriptive Reservoir Model, Network Flow, Linear Programming, Penalty Functions, Depletions, Reservoir Releases, Reservoir Storage, Objective Function, Flood Damage, Navigation, Recreation, Hydropower, Water Supply, Fish and Wildlife

19. ABSTRACT *(Continue on reverse if necessary and identify by block number)*

This report reviews tasks accomplished during Phase I of the project and describes tasks accomplished during Phase II. Phase II tasks include: (1) Additional development of the HEC-Prescriptive Reservoir Model (HEC-PRM), (2) Development of new and enhancement of existing utility programs related to HEC-PRM, (3) Transfer of developed technology to the Missouri River Division, and (4) Preliminary review of procedures for developing system operating rules from HEC-PRM results. The model uses network-flow programming to allocate optimally the system water. In Phase II, the modeled system was extented to include navigation on the Mississippi River by including a node at St. Louis. The applications are performed using the best-currently-available estimates of flow data and penalty functions, both of which must be considered to be preliminary and are basically those used in the Phase I analysis. This report includes draft documentation which describes the current version of the program and selectively displays example input and output.

20. DISTRIBUTION/AVAILABILITY OF ABSTRACT ☒ UNCLASSIFIED/UNLIMITED ☐ SAME AS RPT. ☐ DTIC USERS	21. ABSTRACT SECURITY CLASSIFICATION
22a. NAME OF RESPONSIBLE INDIVIDUAL Darryl W. Davis	22b. TELEPHONE *(Include Area Code)* (916) 756-1104 22c. OFFICE SYMBOL CEWRC-HEC

Missouri River Reservoir System Analysis Model: Phase II

January 1992

Accession For
NTIS GRA&I ☒
DTIC TAB ☐
Unannounced ☐
Justification

By
Distribution/
Availability Codes
Dist | Avail and/or Special
A-1

US Army Corps of Engineers
Hydrologic Engineering Center
609 Second Street
Davis, CA 95616-4687

(916) 756-1104

PR-17

Missouri River Reservoir System Analysis Model: Phase II

Table of Contents

Appendices

List of Figures

List of Tables

Preface

The investigation reported herein is the second of two phases of a project conducted by the Hydrologic Engineering Center (HEC) to assist staff of the Missouri River Division (MRD). The goal of the project is to develop efficient operation plans for the six major reservoirs on the Missiouri River. Phase I began 1 July 1990 and yielded HEC-PRM, a prescriptive model for system analysis. Phase II extends and improves that model.

The project is undertaken in accordance with a task order issued July 1990 by MG Patrick J. Kelly, Director of Civil Works, HQUSACE. HQUSACE point of contact is Earl Eiker, Chief, Hydraulics and Hydrology Branch, Engineering Division, Civil Works Directorate. The project is funded jointly by MRD, the National Drought Study, and the Civil Works Research and Development program.

The project is a joint effort by HEC and the Institute for Water Resources (IWR). IWR staff are responsible for development of cost based penalty functions critical to the analysis. Reports of their work are published separately.

Mike Burnham, Chief, Planning Analysis Division, HEC, is project engineer. Bob Carl, Planning Analysis Division, HEC, developed the software and performed applications described herein. Other HEC staff assisting include Vern Bonner, Marilyn Hurst, and Richard Hayes. Darryl Davis, Director, HEC, provided general supervision and guidance for the project.

HEC staff were assisted by the following consultants in Phase II of this project: Prof. Paul Jensen, University of Texas at Austin; Dr. David Ford, Consulting Engineer; and Prof. Jay Lund, University of California, Davis.

Chapter 1

Executive Summary

The six main-stem reservoirs of the Missouri River system are operated by the Corps of Engineers for flood control, navigation, irrigation, power, water supply, water quality control, recreation, and fish and wildlife protection. Changes in supplies and changes in demands motivated a review of system operation policy by MRD staff.

To provide information for decision making in this review, HEC developed a prescriptive model. This model, designated HEC-PRM, represents the system as a collection of nodes and links and uses network-flow programming to allocate optimally the system water for competing purposes. This approach was selected because it satisfies institutional, economic, environmental, and engineering criteria. Goals of and constraints on system operation are represented with penalty functions. The functions are both cost-based and non-cost-based. The cost-based functions are developed by evaluating economic cost incurred or the value of opportunity foregone. The non-cost based functions were developed to reflect environmental outputs and concerns, regional priorities on type and location of outputs, and risk-management objectives. The model is implemented on a personal computer running MS-DOS. It makes extensive use of HEC's data storage system (HEC-DSS). Prior to application as a decision-making tool in the review, the model was validated subjectively by comparing prescribed operation for a historical period.

Model development and testing constitute Phase I of HEC activities. Phase II of the activities began in January 1991. Table 1 summarizes the proposals and accomplishments for Phase II of the

Table 1
Phase II Activities and Accomplishments

Proposal	Accomplishment
Expand the system analyzed	Model extended to St. Louis and expanded to full hydrologic record.
Refine the penalty functions used	Refinement generally lagged development. Flood control issue addressed.
Improve HEC-PRM's user interface	Character-based interface implemented. This provides additional display and analysis tools via MATHPK, DSPLAY, PRMPOST.
Make technical improvements to HEC-PRM	Significant improvements to network solver. Hydropower algorithm formulated.
Complete additional Phase II analyses	Delayed due to unanticipated delays in development of penalty functions.
Transfer developed technology to MRD staff	Workshop held. Variety of reports, documents prepared.

Missouri River System Analysis.

As an extension of the Phase II activities, HEC is reviewing alternatives to enable recommendation of procedures for developing system operating rules from HEC-PRM results. This task is necessary to bridge the gap between theory (the prescriptive model) and practice (the real-world system). Results are forthcoming.

In calendar year 1992, HEC staff plan improvements to HEC-PRM for other on-going studies. These include, but are not limited to, the following:

(1) further refinement to the user interface;
(2) integration and testing of the improved network solver; and
(3) implementation and testing of the hydropower algorithm.

The goal is to develop a general-purpose model and software that can be applied on any system. As HEC-PRM is improved, updates will be provided to MRD.

Chapter 2

Hydrologic Engineering Center Prescriptive Reservoir Model

Motivation for Model Development

System Description. In the April 2, 1991 issue of the Wall Street Journal, staff reporter Dennis Farney wrote the following about the Missouri River:

> *Until recent decades, it was known as "Old Misery" - a treacherous stream... But today, the Missouri is a blue-collar river: harnessed, dammed and channelized for flood control, hydroelectric power, barges, and irrigation. It can generate 2,400 megawatts of electricity [and] hold enough water to supply Kansas City for the next two centuries.*

The Missouri River main-stem reservoir system consists of six reservoirs. Figure 1 shows the location of these reservoirs and the major streams in the system. Pertinent characteristics of the reservoirs are shown in Table 2. Authorized purposes for these reservoirs include flood control,

Table 2
Missouri River Reservoir Characteristics

Reservoir	Drainage Area (sq mi)	Avg. Total Inflow, (cfs)	Reservoir Capacity (1000 acre-ft)	Total Discharge Capacity (cfs)	Shoreline Length, (mi)	Avg. Annual Energy (10^6 kWh)
Ft. Peck	57500	10200	18688	291000	1520	1044
Garrison	181400	25600	23923	796000	1340	2354
Oahe	243490	28900	23338	245000	2250	2694
Big Bend	249330	28900	1874	373000	200	1001
Ft. Randall	263480	30000	5574	680500	540	1745
Gavins Point	279480	32000	492	381000	90	700

navigation, irrigation, power generation, water supply, water-quality control, recreation, and fish and wildlife protection.

The Missouri River system operation is based on decreasing priority for the following purposes:
(1) flood control;
(2) all irrigation, and other upstream water uses for beneficial consumptive purposes;
(3) downstream M&I water supply and water quality requirements;
(4) equitable service to navigation and power;
(5) efficient generation of power to meet the area's needs; and
(6) "insofar as possible without serious interference with the foregoing functions.." benefit to recreation, fish and wildlife (USACE, 1979).

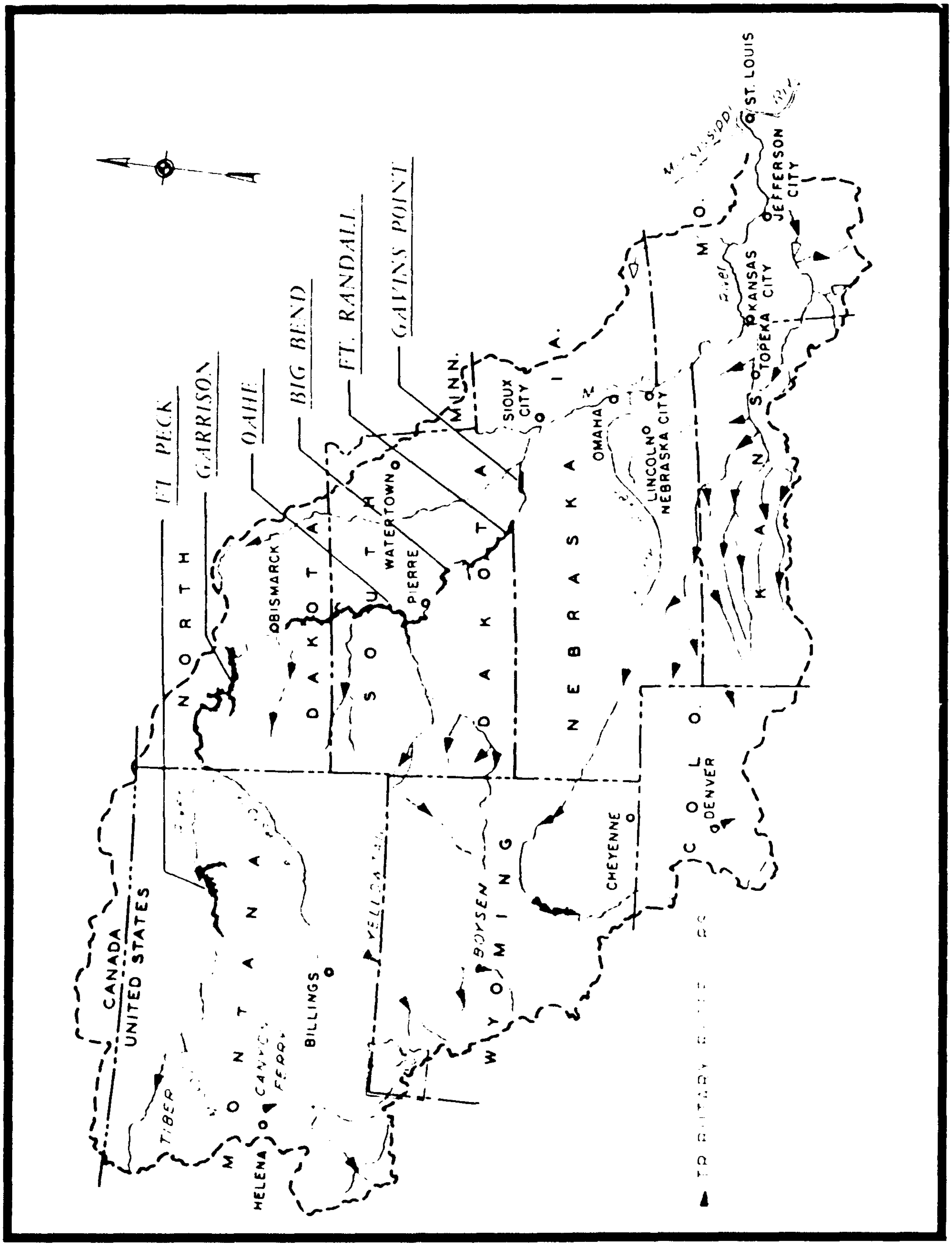


Figure 1
Missouri River Basin and Major Dams

Operation Problems. A five-year drought in the midwestern United States has strained the Missouri system. In early 1991, the reservoirs were at near-record lows. Oahe reservoir, for example, held less than 60% of its 1986 peak storage; the reservoir water surface was 28 ft below the 1986 level. As a consequence, upstream users were forced to compete with downstream users for the reduced supplies. The recreation industry that developed in North Dakota, South Dakota, and Montana protested as the system operator, the Corps of Engineers, released water from the reservoirs for downstream navigation.

Missouri River Main Stem Plan of Study. To provide information for making decisions regarding changes in Missouri River mainstem operations, the Corps initiated in 1989 a formal review of the Master Water Control Manual, the basic document for operating the system. The Corps formed a multidisciplinary study team within the Missouri River Division (MRD) office and adopted a formal plan of study (USACE, 1990a). The goal of the study is to determine if the current plan or an alternative plan for Missouri River mainstem operations best meets the current needs of the basin. To do this we need to identify the economic, environmental, and social benefits of satisfying competing demands. Study team staff accepted early that analytical tools would play a key role in the review, and set out to identify one or more appropriate reservoir-operation analysis tools. Those considered may be classified broadly as descriptive or prescriptive. Yeh (1985) has provided an extensive review of both categories.

A *descriptive* tool would be used to answer the question "How would the system perform if we followed this policy or set of priorities?" The alternative policies considered are proposed by a user. MRD study staff proposed to conduct the study with such a tool. The study procedure proposed initially relied on L01, the MRD long-range regulation simulation program (USACE, undated). In the first phase of the study,

> *... the operation of the main stem reservoir system will be simulated over the period of record from 1898 to the present to provide a base line condition. This base line condition will be analyzed in hydrologic, economic, and environmental terms to identify issues and conflicts. Alternative water control plans ... will be formulated and evaluated in hydrologic, economic, and environmental terms. The evaluation of these alternatives will identify which of these plans favor each of the main stem project uses* (USACE, 1990a).

In the second phase, MRD staff proposed to use the descriptive model to evaluate promising alternatives in further detail.

A *prescriptive* tool, on the other hand, would be used to answer the question "How should we operate the system if we accept this definition of the goals of and constraints on system operation?" A prescriptive tool generates iteratively the alternative policies to be considered and evaluates the feasibility of each with a built-in simulation model. It quantifies the efficiency of each feasible alternative using a formal definition of operation goals and objectives,. Finally, after evaluating all alternatives, it identifies the best policy. Examples of prescriptive tools are linear-programming models, nonlinear-programming models, and dynamic-programming models.

To promote systematic comparison of the alternative models, a set of critical questions was posed. These are summarized in Table 3. Based on consideration of these, HEC proposed to develop and apply a prescriptive model for the Missouri River main stem study (USACE, 1991b). The most compelling argument for proposing a prescriptive model to assist in studies was that it would provide a highly efficient means for developing reasonable alternative policies to represent accurately the

Table 3
Criteria for Model Selection

Criteria	Questions
Institutional Criteria	Will the model provide directly information that will help solve the system-operation problem?
	Can the model represent all system operation purposes accurately?
	Can the model evaluate alternative priorities for system operation?
	Can the model outputs be translated into terms that are readily understandable to users?
	Can the model be modified or expanded easily as more information becomes available, as understanding of the system operation improves, and as the users become more sophisticated?
	Can the model be used on the computer hardware available to users?
	Can the model be implemented in time to provide information for decision making?
Economic Criteria	Can the model evaluate accurately the economic impact of operation decisions?
	Can the economic data required for the model be obtained with reasonable effort?
Environmental Criteria	Can the model treat non-quantifiable operation purposes, such as fish and wildlife protection?
	Can the model represent adequately the requirements for endangered species?
Engineering Criteria	Does the model use existing data or data that can be obtained with reasonable effort?
	Can alternative future inflow or demand sequences be studied conveniently?
	Can results from the model be used to evaluate risk?
	Is the technology embodied dependable?
	Is the model fast enough to permit analysis of many alternatives?

competing demands.

Prescriptive Model Description

Mathematical Formulation. The prescriptive model developed by HEC addresses the Missouri River water allocation problem by:

(1) representing the physical system as a network;

(2) formulating the allocation problem as a generalized minimum-cost network-flow programming problem [see Jensen and Barnes (1980) for details of network-flow programming];

(3) developing, for various users or purposes, an objective function that represents desirable

operation for that user or purpose;

(4) solving the network problem with an off-the-shelf solver; and

(5) processing the network results to define, in convenient terms, system operation.

The network representation of the reservoir system is similar to the approach used by Sigvaldason (1976), Martin (1982), Ikura and Gross (1984), Sabet, et al. (1985), and Chung, et al.(1989). The network arcs represent any facilities for transfer of water between points in time or space. For example, a natural channel transfers water between two points in space and is represented by an arc. A reservoir transfers water between two points in time; this transfer, too, is represented by an arc of the network. Network arcs intersect at nodes. Flow is conserved at each node. The nodes represent river or channel junctions, gage sites, monitoring sites, dams, or water-demand sites. To analyze multiple-period system operation, each site in space is represented by a node for each time period.

Penalty Functions. Central to the proposed approach (and, for that matter, to use of any prescriptive model) is the idea that goals of and constraints on system operation can be represented with objective functions. The difficulty in doing so is well recognized. Liebman (1976) wrote of this that "...human tastes are different...even with perfect knowledge of all effects, interactions, and implications, individual definitions of Utopia would be different...even if there were complete agreement on goals, ...there would still be disagreement over which is the best way, and even on what criteria should be used to define best."

These differences in preferences and measures of success were well understood by the study team. However, the study goal is to provide information to be used to define the best operation, not to develop a model that would define the best operation. The MRD study team and HEC felt that a prescriptive model with alternative objective functions, representing various goals and various rankings of goals would provide much of this information. For example, a purely economic objective function with inviolable constraints to protect lake recreation will identify the cost of such constraints.

For this study, cost-based penalty functions were developed by MRD staff in cooperation with the IWR. The current status of these functions is described in Chapter 3. The functions developed can be of two types: cost-based or non-cost-based. The cost-based functions, "...show the loss in economic value as the flow in each model link deviates from the optimum flow (USACE, 1990b)." Individual economic cost-based penalty functions were developed for the following outputs: urban and agricultural flooding; water supply; recreation; hydropower; and navigation. The functions relate cost to either reservoir release, reservoir storage, or channel flow at downstream locations. The functions vary by month if appropriate. At this time, non-cost-based penalty functions have not been used in this study. If utilized, they can represent goals of system operation that cannot be quantified in economic terms. For example, a flow requirement for fish and wildlife protection may be represented with a penalty function in which the penalty arbitrarily is set to force the desired operation. As with the cost-based functions, these functions relate penalty to either reservoir release, reservoir storage, or channel flow at downstream locations and vary by month if appropriate.

For analysis of system operation, the penalty functions are summed. The non-cost based functions are believed to be in units commensurate with the cost-based functions, enabling the functions to be summed. Otherwise, the functions can be treated separately. The resulting functions are represented in a piecewise-linear fashion for the network model. This manipulation of penalty functions is described in further detail in Chapter 3.

Software. Software to implement the proposed solution is general purpose. Figure 2 shows schematically the components. These include software to:

(1) formulate the network;

(2) solve the minimum-cost generalized network;

(3) manage the data; and

(4) display and help analyze the results.

HEC's Prescriptive Reservoir Model, HEC-PRM, formulates the minimum-cost network-flow programming problem that represents the system operation problem (USACE, 1991c). From the user's description of system reservoir storage, release facilities, interconnecting channels, diversions, and hydropower facilities, the program defines the network layout. HEC-PRM accesses flow data and penalty functions stored in data bases to define the parameters of the network arcs.

Several alternative solvers are available, depending on the level of detail of representation of the reservoir system. If evaporative losses are negligable, an out-of-kilter solver (Ford and Fulkerson, 1962) can be used to solve the resulting pure network problem. Otherwise the problem requires a generalized network solver. Initially, the dual flow-augmentation solver developed by Jensen and Bhaumik (1974) was used. This will be replaced with an improved solver developed by Jensen (1991), as discussed in Chapter 5 of this report.

To manage the mass of time-series and relational data required, HEC's data storage system, HEC-DSS (USACE, 1990c), is used. The software provides a systematic means for organizing, storing, retrieving, manipulating, and sorting data. Its capabilities are accessible via FORTRAN. Program DSPLAY graphs any of the data stored with HEC-DSS, including inflow time series, penalty functions, and, after solution of the minimum-cost network problem, the prescribed release, storage,and flow time series. A variety of utility programs manipulate and preprocess data necessary for the analysis. For example, a utility program converts penalty functions provided in spreadsheet format to the HEC-DSS format. Another computes the parameters of a piecewise-linear approximation of a nonlinear penalty function. These programs are described in further detail in Chapter 3.

Missouri System Representation with Model. The network representation of the Missouri River main stem system represents Ft. Peck, Garrison, Oahe, Big Bend, Ft. Randall, and Gavins Point reservoirs and non-reservoir control points Sioux City, Omaha, Nebraska City, Kansas City, Boonville, and Hermann. This system is shown by Figure 3. Inflows are specified for each control point. Lake evaporation is described as a function of storage for each reservoir, and is represented in the network with the arc multiplier. The present model was extended to St. Louis, Missouri, on the Mississippi River.

Penalty functions are defined for storage in each reservoir, for release from each reservoir, and for flow at each of the non-reservoir control points. Additional penalty functions for hydropower are defined as a multivariate function of head and release at each reservoir. For initial analysis, these were simplified by assuming constant head. Subsequently, a successive linear programming algorithm similar to that proposed by Grygier and Stedinger (1985) will be implemented to eliminate this assumption. This algorithm is described in further detail in Chapter 3.

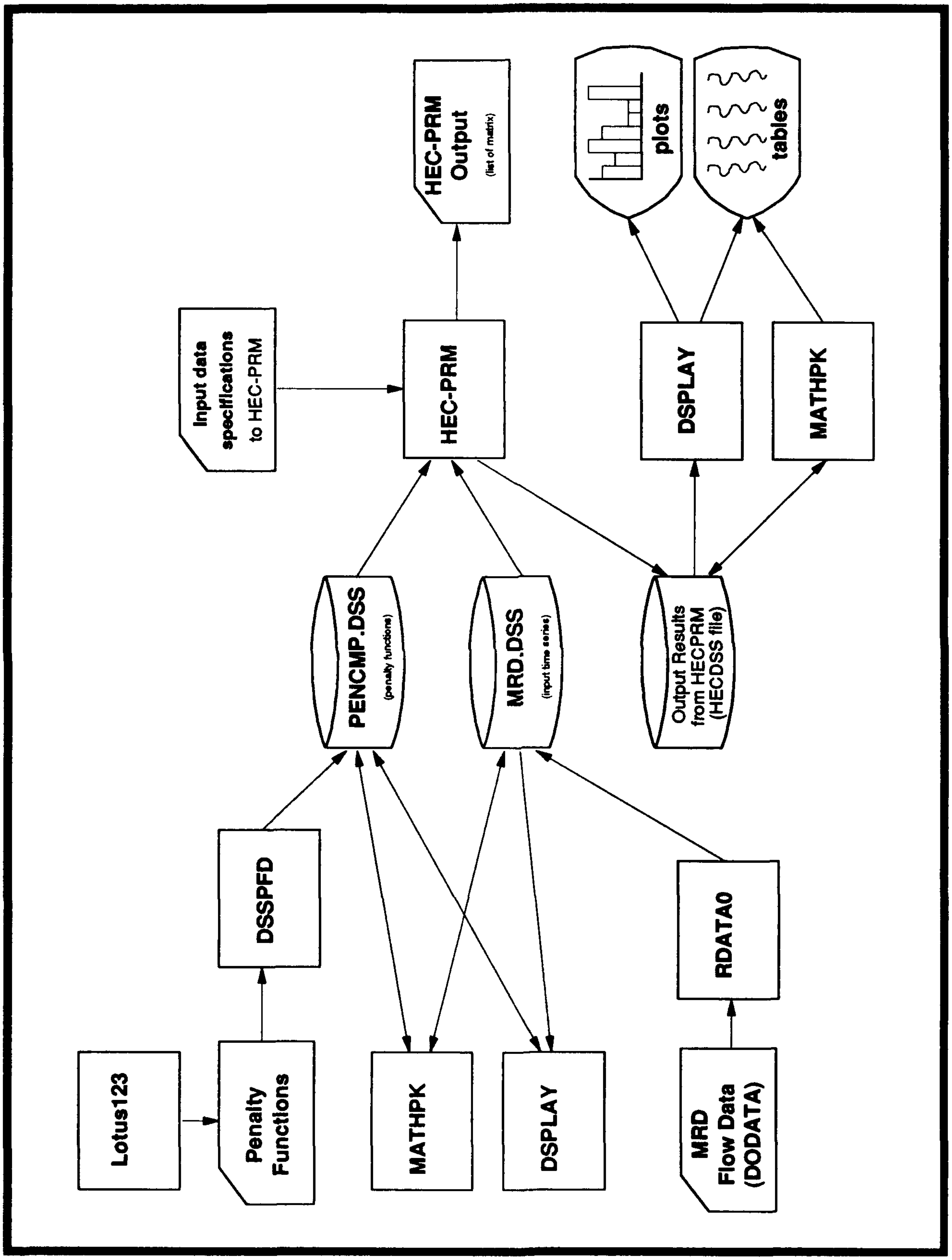


Figure 2
Components of Software

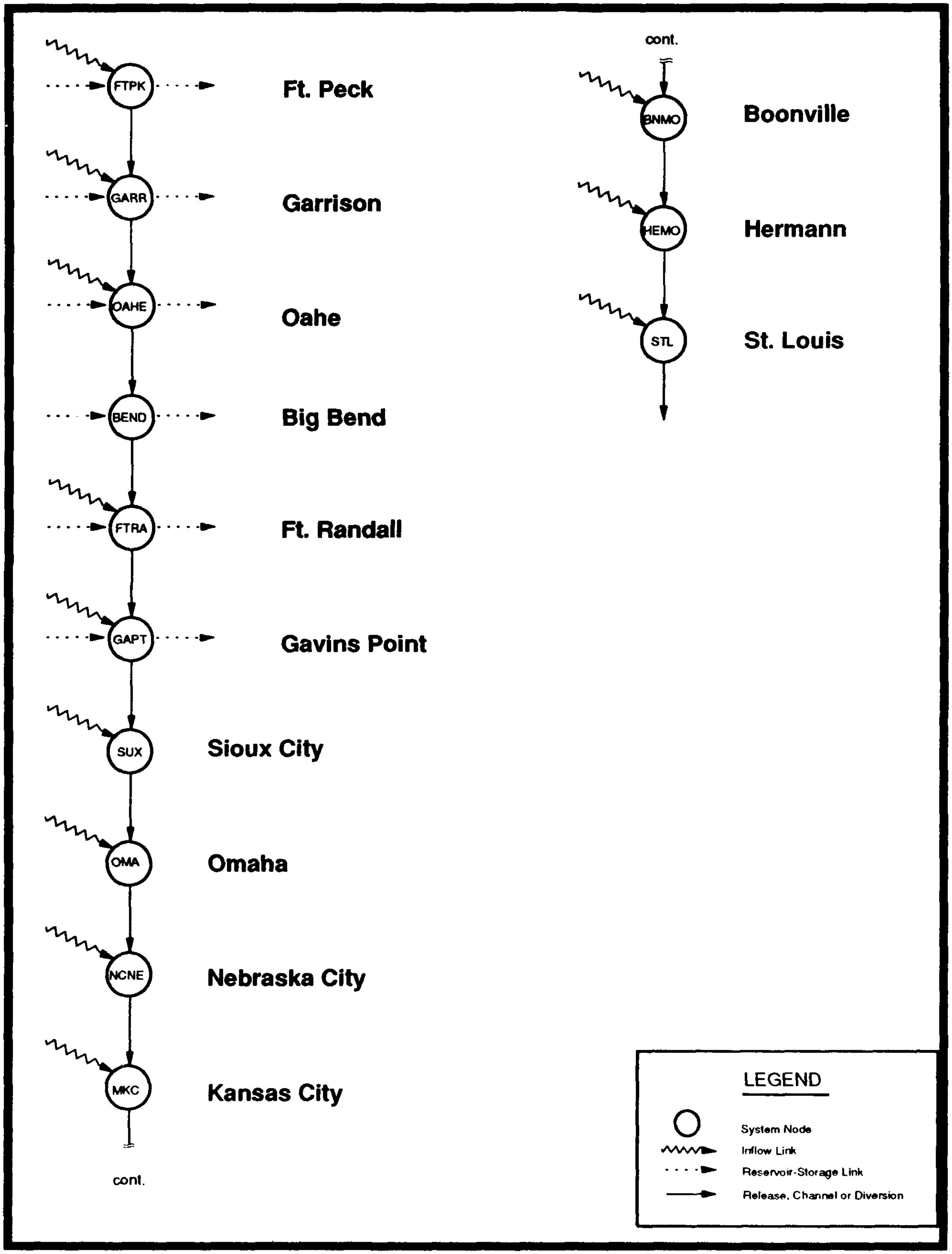


Figure 3
Single-period Network Representation of System

Phase I Applications of HEC-PRM

HEC-PRM was applied by HEC staff for preliminary analysis of the Missouri River system in Phase I of this study. These results were presented in an earlier report (USACE, 1991b) and are summarized here for completeness.

Validation of Model. Unlike a descriptive model, HEC-PRM cannot be validated directly by comparison with historical data. No such data can exist because the objective function used in the model never reflects exactly all goals of and constraints on operation. Consequently, historical operation never is truly optimal for the objective function used in the model. However, HEC staff felt that properly-defined, current-condition, aggregate penalty functions would correlate highly with current operation rules because those functions have developed around expected operation. For example, marina owners build boat docks where they expect the lake level to be. They would incur no cost if the reservoir is operated to hold the lake at that level. Their penalty function would reflect this. The marina owner's desired level, over time, becomes the level corresponding to operation with the current policy. Similar arguments can be made for facilities to withdraw irrigation water from reservoirs, and for works to divert water for supply in downstream reaches. Thus operation to minimize the penalty should reflect, to a large degree, current operation rules. This argument led to expecting the operation proposed by the model with current-condition penalty functions to compare favorably with current operation. A test was thus devised.

For the test, Missouri River system operation for March 1965 to March 1970 was analyzed with HEC-PRM. Best estimates of current-condition penalty functions were used. The resulting network included 5327 arcs and 1142 nodes. The results were compared with operation following current rules, determined with L01. A perfect match of operation was not expected. Indeed, the results should not be identical, as the models employ different simplifications of the real system, and the descriptive model does not have the foresight of the prescriptive model. Computed reservoir storages for Ft. Peck, Garrison, Oahe, and Ft. Randall for the validation period are shown in Figure 4 through Figure 8. For simplicity, storage is shown in thousand acre-feet (KAF). The pattern of storage indicated by the two models for Ft. Peck (Figure 4) matches well. The seasonal cycles are identical. HEC-PRM proposes slightly less storage for 1967-1969. Some slight differences in the storages are attributable to the approximation of the evaporation. At lower storages, the evaporation is over-estimated. This is true for all reservoirs. The general pattern of storage for Garrison, shown by Figure 5, also matches well. The storage indicated by HEC-PRM from mid-1966 to early 1968 is about 20% less than that indicated by the MRD model. In that same period, HEC-PRM has proposed greater release due to the energy penalty function: The advantage of releasing water for energy exceeds the advantage of storing it in these months, even for future use. Thus HEC-PRM draws down the reservoir. The overall Oahe storage pattern, shown in Figure 6, follows the pattern of the MRD model. Figure 7 shows storages at Ft. Randall proposed by the two models. These match well, with what appears to be a slight time lag in the HEC-PRM results. This time lag is the result of the capability of HEC-PRM to incorporate knowledge of future inflows in making release decisions: If postponing releases will reduce the overall penalty, HEC-PRM will do so. Figure 8 shows the total storage in the mainstem reservoirs which consists of Ft. Peck, Garrison, Oahe, Big Bend, Ft. Randall, and Gavins Point.

As a consequence of this validation study, HEC and MRD staff accepted HEC-PRM as a tool for analysis of the Missouri system.

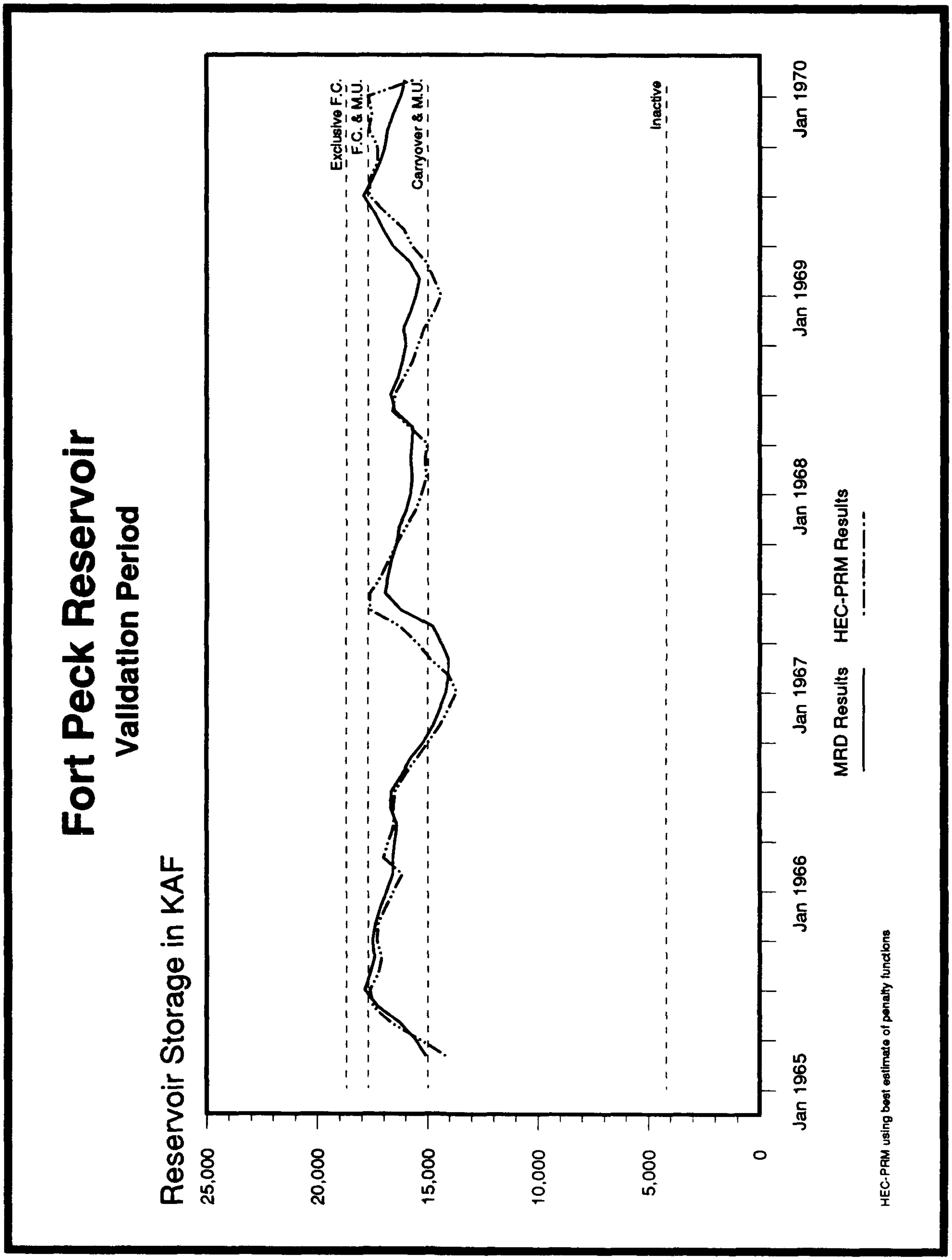


Figure 4
Ft. Peck Storages Prescribed in Validation Test

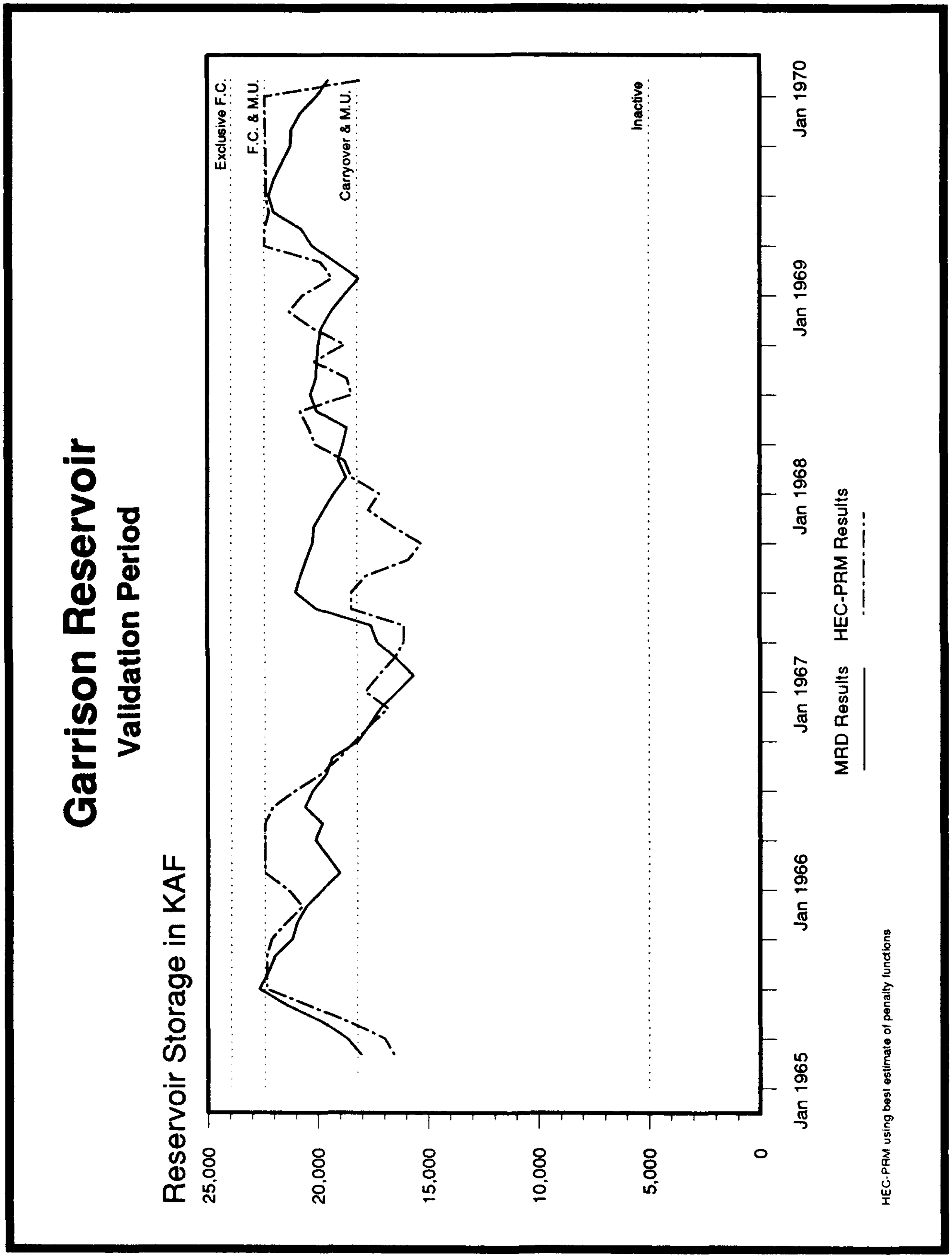


Figure 5
Garrison Storages Prescribed in Validation Test

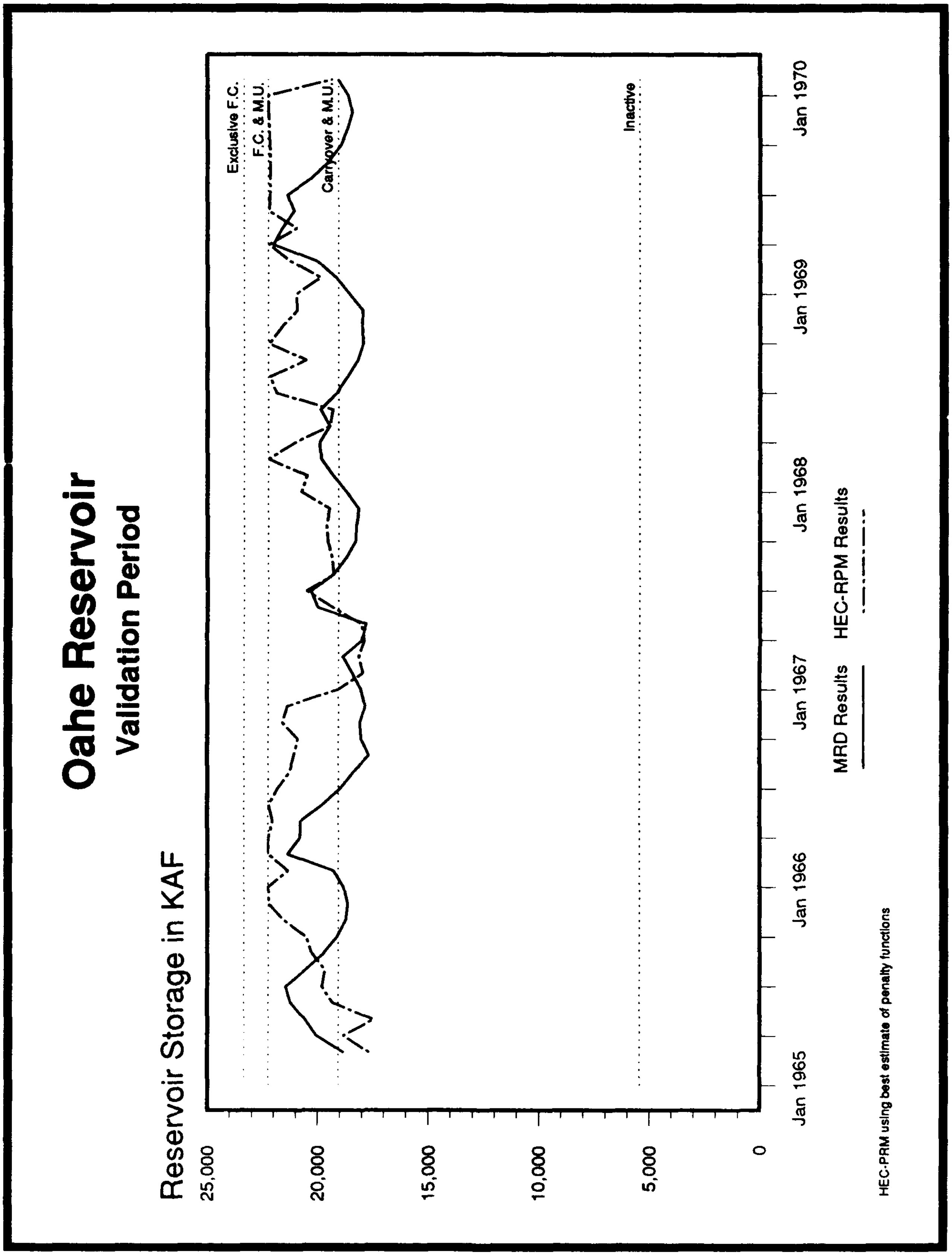


Figure 6
Oahe Storages Prescribed in Validation Test

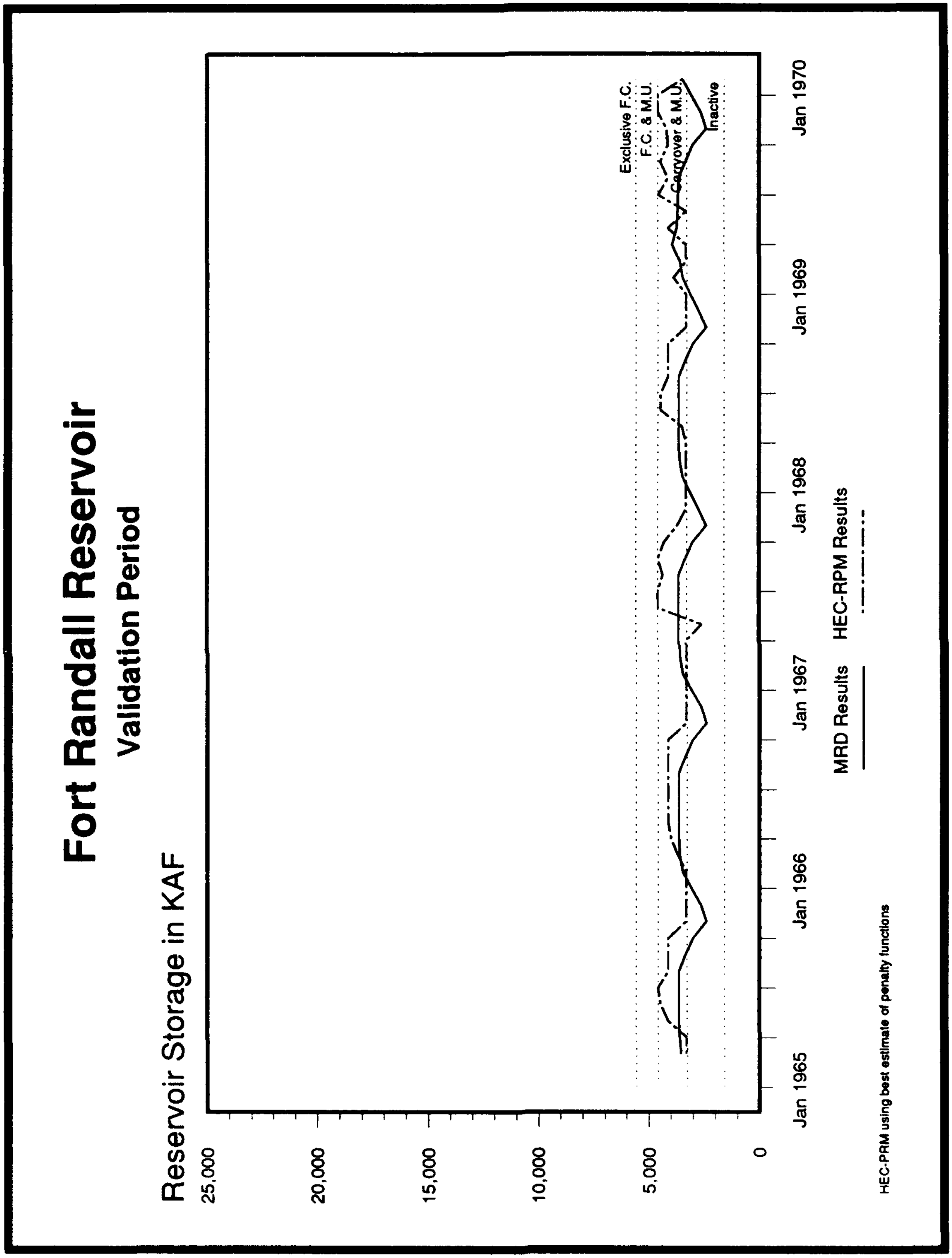


Figure 7
Ft. Randall Storages Prescribed in Validation Test

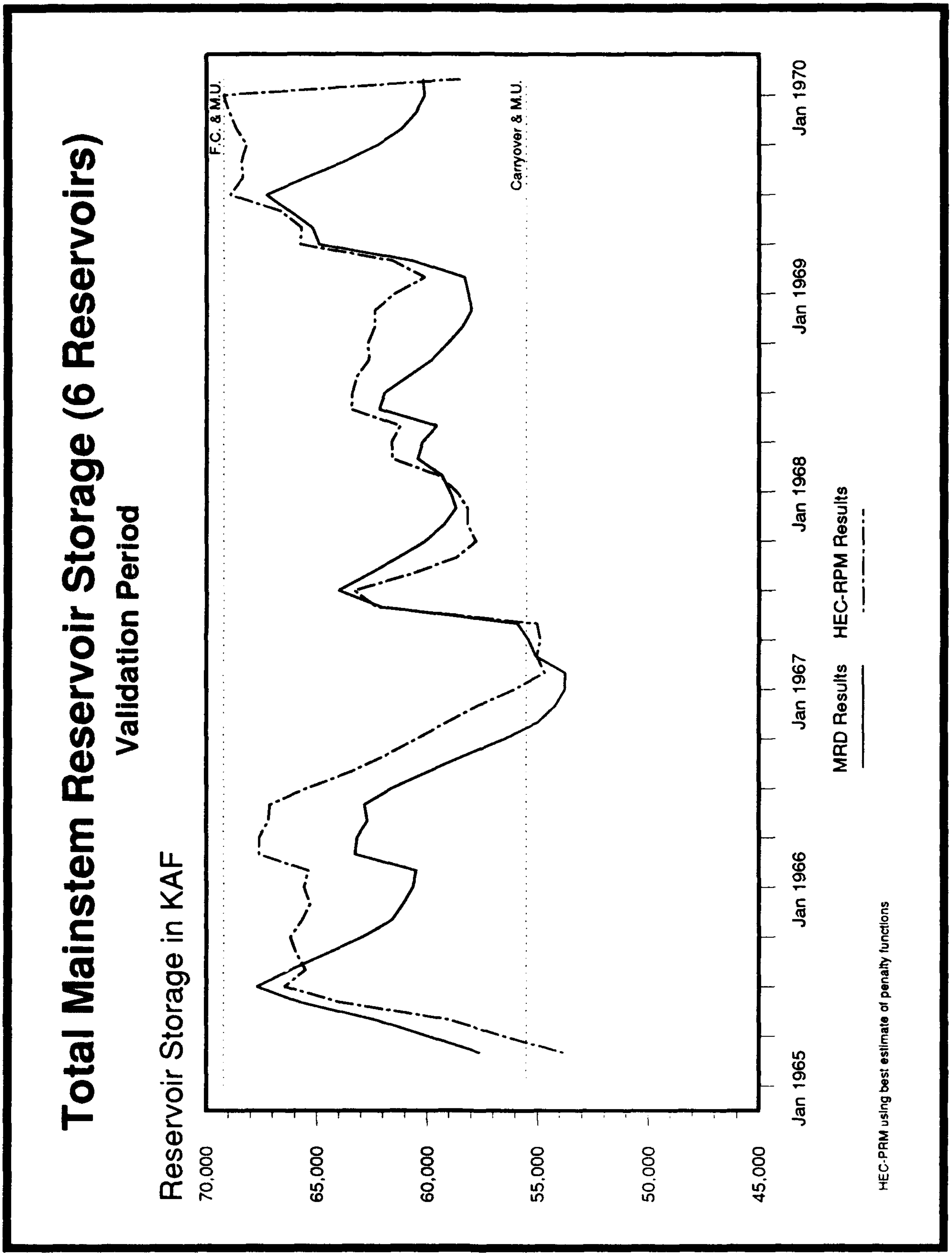


Figure 8

Total Storage in Mainstem Reservoirs Prescribed in Validation Test

Phase I Applications. Phase I applications of HEC-PRM included the following:

(1) analysis of the system critical period, March 1930 - March 1953, with the best-currently-available estimates of system penalty functions; and

(2) analysis of the same period with a hypothetical navigation penalty function.

Chapter 3 of this report describes further applications of the model.

Critical Period with Best Currently-Available Penalty Functions. The network required to represent system operation for this critical period included 23375 arcs and 4970 nodes. Figure 9 through Figure 13 show the HEC-PRM proposed end-of-month storages at Ft. Peck, Garrison, Oahe, and Ft. Randall for the critical period. As a rule, energy generation dominates the operation. HEC-PRM proposes release of water to drive the energy penalty to zero if sufficient water is available. Otherwise, it proposes making no release and storing water for subsequent use. This is a case of farsighted versus shortsighted decision making. The model must choose between making minimum releases for hydropower now or storing water for later use. It choses the latter based on system penalty for the total period of analysis, as defined by the penalty functions. Although a skilled operator might choose a smoother operation scheme, the penalty functions used in this application do not indicate that another policy is better, although it may be as good.

Figure 14 shows channel flows at Sioux City and Kansas City if the system is operated according to the policy found by HEC-PRM. At Sioux City, the penalty would be great if the flow is less than approximately 500 KAF/month (8,260 cfs) in January or less than 1600 KAF/month (26,400 cfs) in the remainder of the year. HEC-PRM has proposed releases that will meet this minimum. At Kansas City, the penalty would be great if the flow is less than 500 KAF/month (8,260 cfs) in January or 2200 KAF/month (36,400 cfs) in the remainder of the year. Again, HEC-PRM has proposed releases to meet this minimum. Similarly, HEC-PRM has proposed releases that will limit the channel flow at Sioux City to well below the discharge at which penalty again is great. This is 2500 KAF/month (41,300 cfs) in January or 8000 KAF/month (132,000 cfs) in the remainder of the year. In fact, most flows are in the range between the desired minimum and the desired maximum, thereby incurring little or no penalty. The same is true at Kansas City. The flow is frequently in the range 500 or 2200 KAF/month (8,260 or 36,400 cfs) to 3600 KAF/month (59,500 cfs). The flow at Kansas City is clearly outside this range in 1947 and again in 1951. However, reservoir operation could do little to reduce these extreme flows, as they are the consequence of uncontrollable local inflow. For example, when the flow at Kansas City reaches approximately 10000 KAF/month (174,000 cfs) in 1951, the local inflow between Nebraska City and Kansas City is almost 9000 KAF/month (149,000 cfs).

Critical Period with Hypothetical Navigation Penalty Function for Sioux City Flow. In the second Phase I application of HEC-PRM, a hypothetical navigation penalty function was added at Sioux City to demonstrate the impact of system operation for high-penalty downstream requirements. This function increased by approximately an order of magnitude the unit penalty for flow outside the range desired for navigation (1875 - 7200 KAF/month or 31,000 - 119,000 cfs during the months April to November). Figure 15 through Figure 19 show the prescribed storages for Ft. Peck, Garrison, Oahe, Ft. Randall, and total mainstem storage for the critical period using this hypothetical navigation penalty function.

The hypothetical navigation penalty function causes the flow pattern at Sioux City to be smoother, as the range of flows there is reduced. Figure 20 shows the downstream flows at Sioux

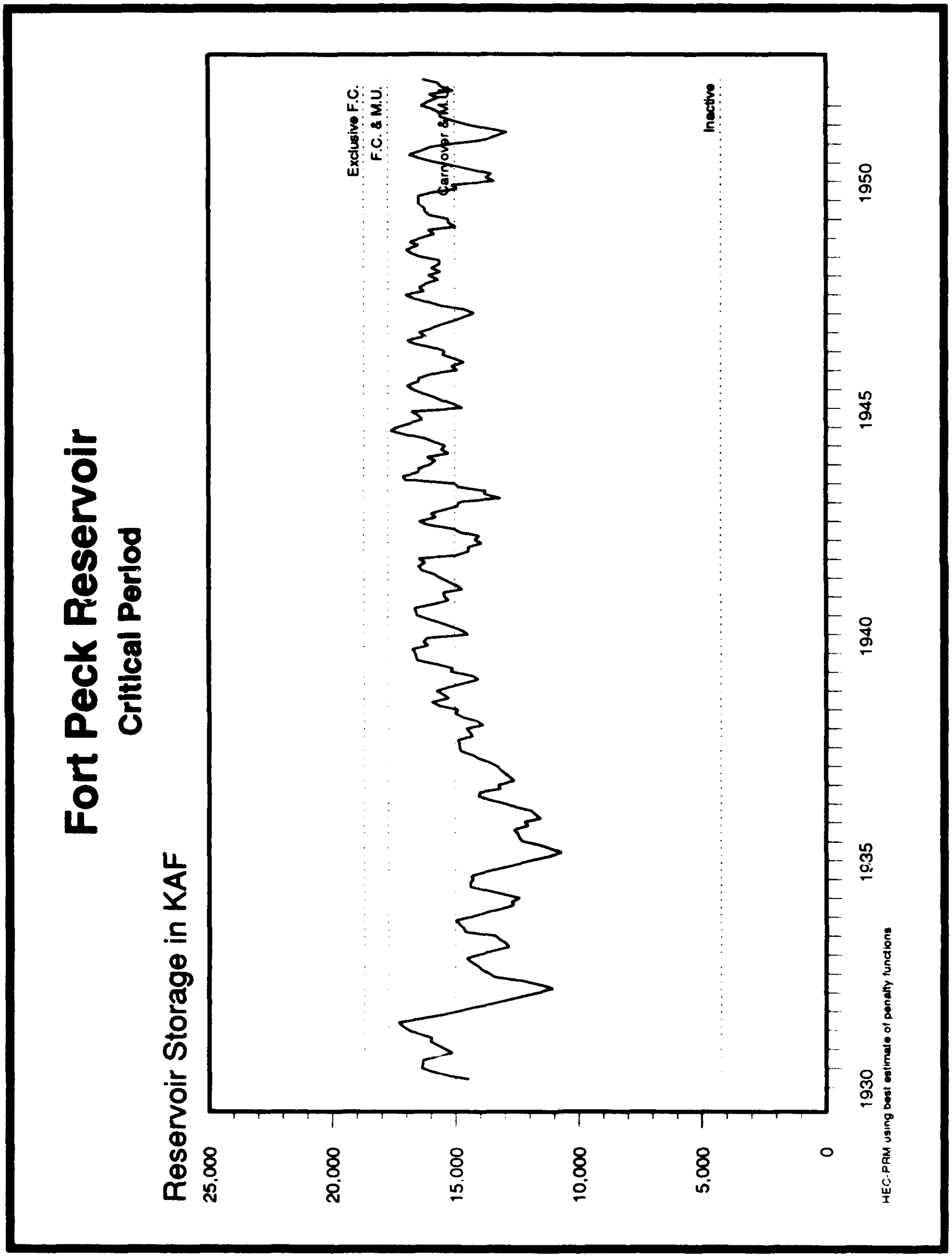


Figure 9
Fort Peck Prescribed Storages for Critical Period with Best Estimate Penalty Function

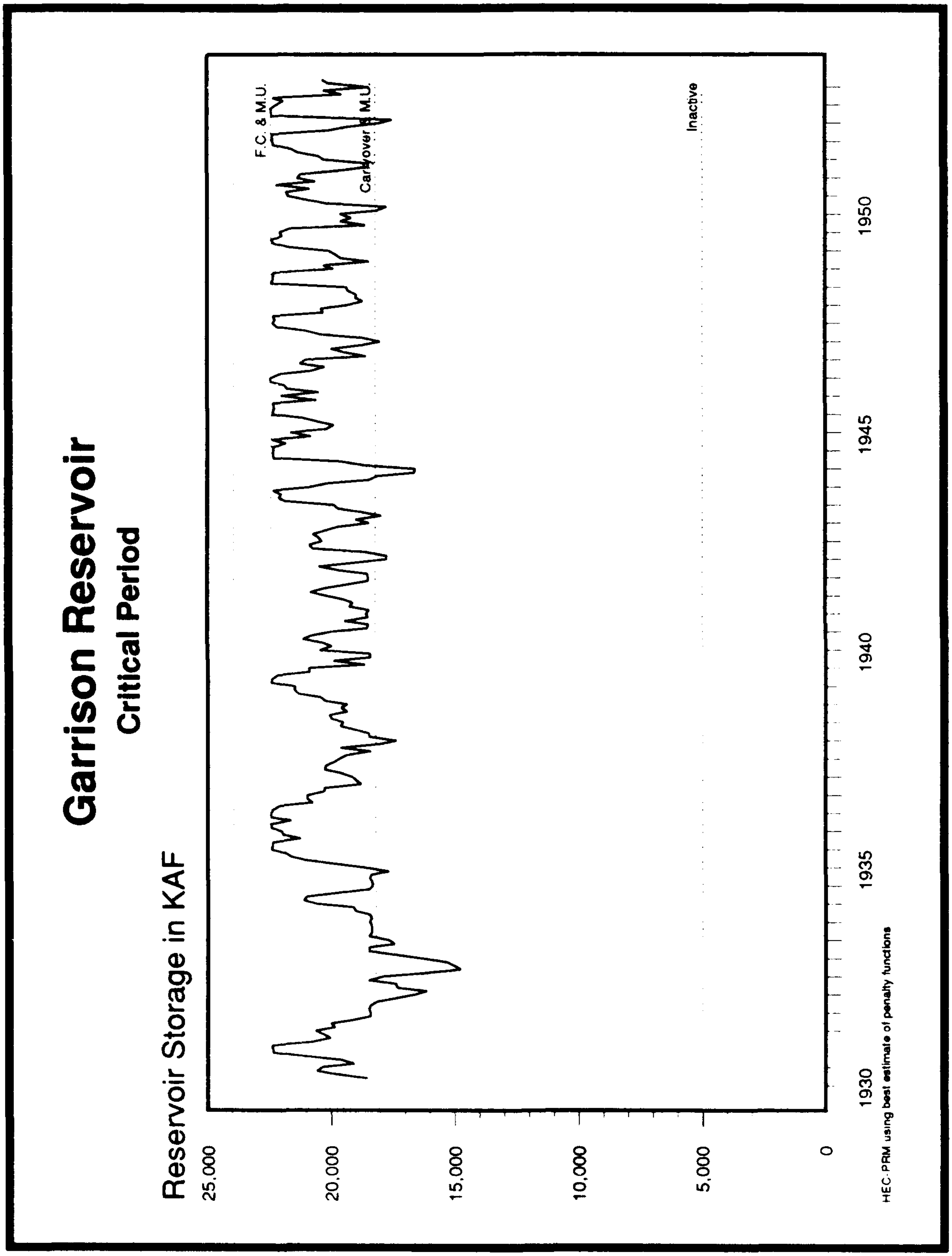


Figure 10
Garrison Prescribed Storages for Critical Period with Best Estimate Penalty Functions

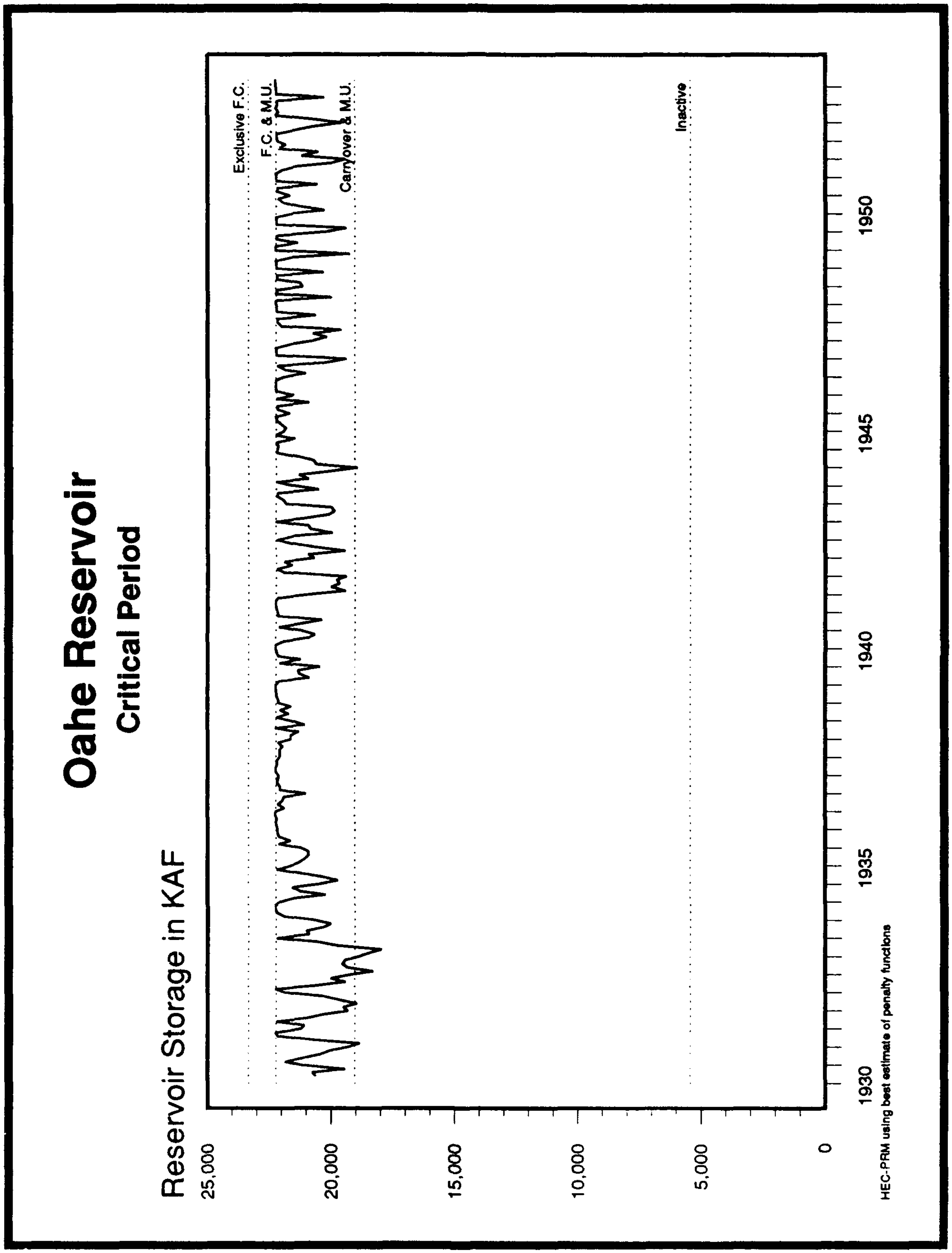


Figure 11
Oahe Prescribed Storages for Critical Period with Best Estimate Penalty Functions

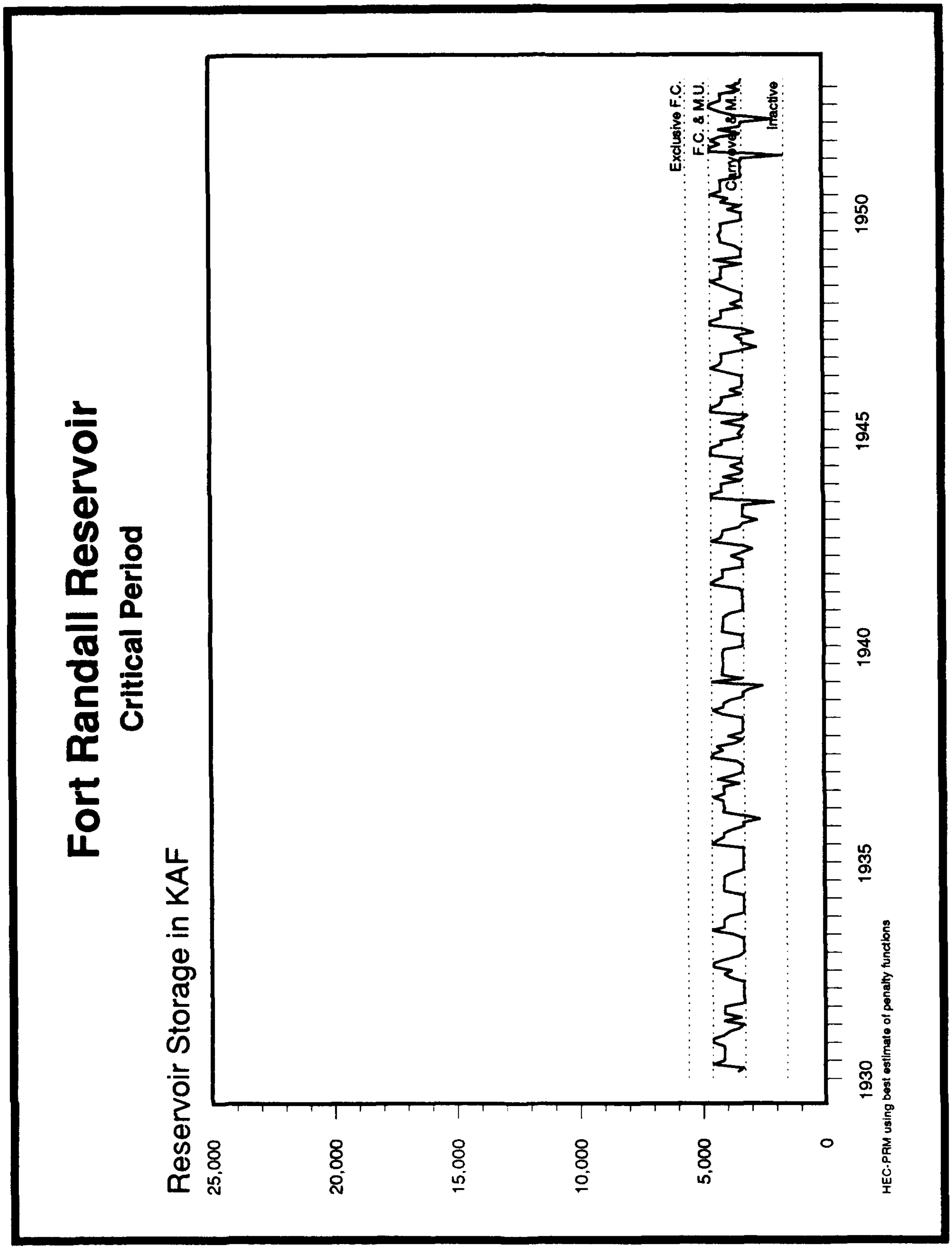


Figure 12
Ft. Randall Prescribed Storages for Critical Period with Best Estimate Penalty Functions

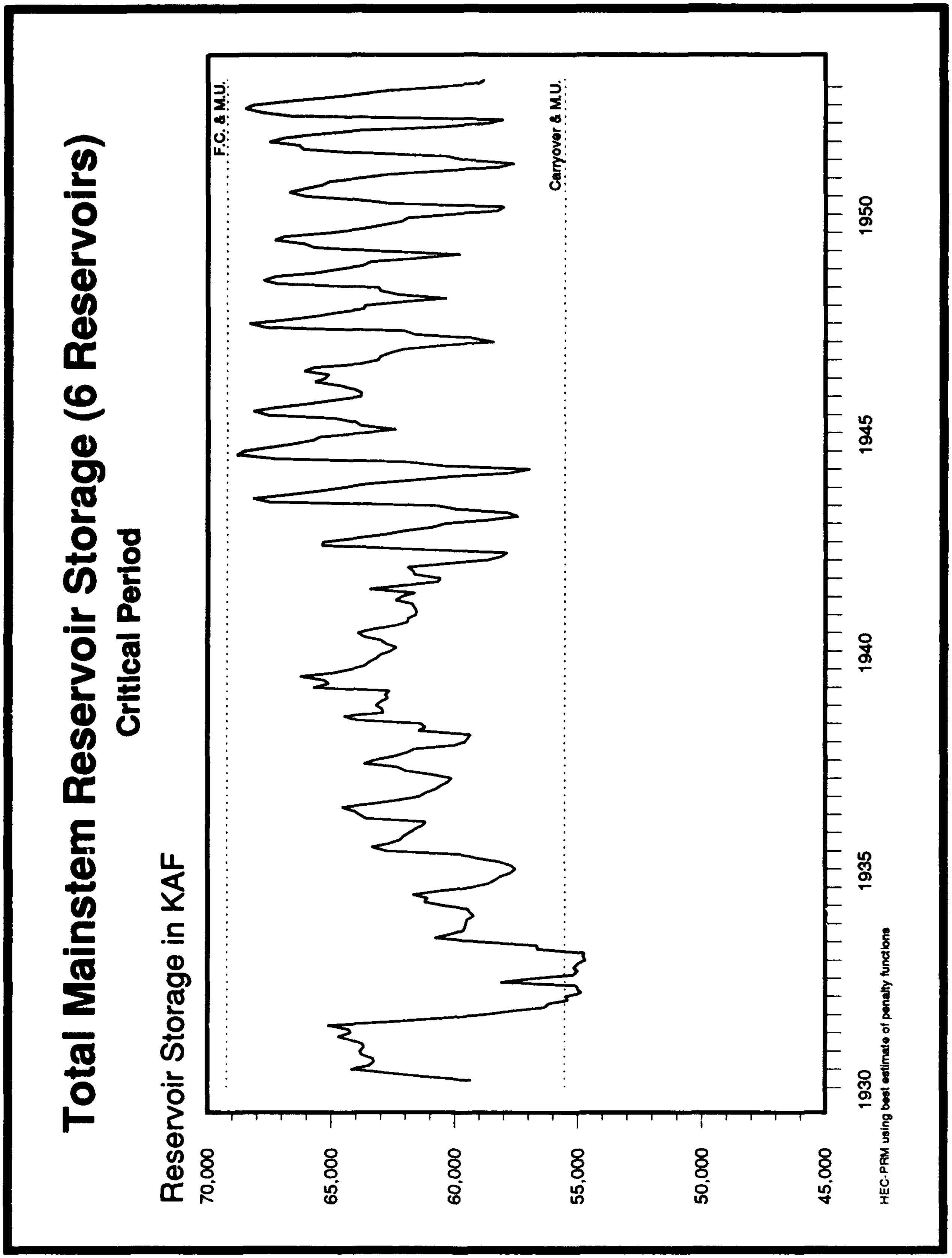


Figure 13
Total Peck Storages Prescribed for Critical Period with Best Estimate of Penalty Function

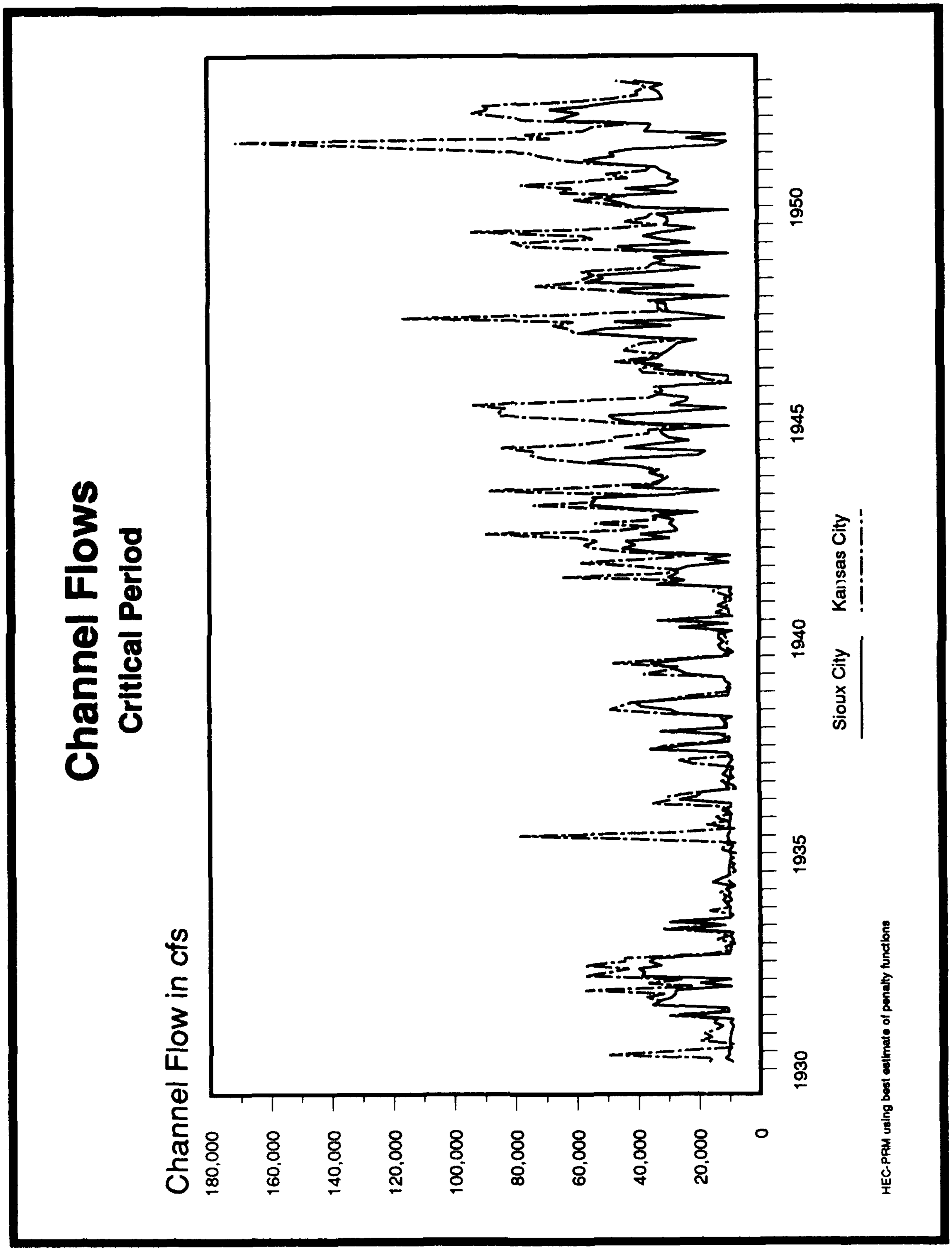


Figure 14
Sioux City and Kansas City Flows for Critical Period with Best Estimate Penalty Functions

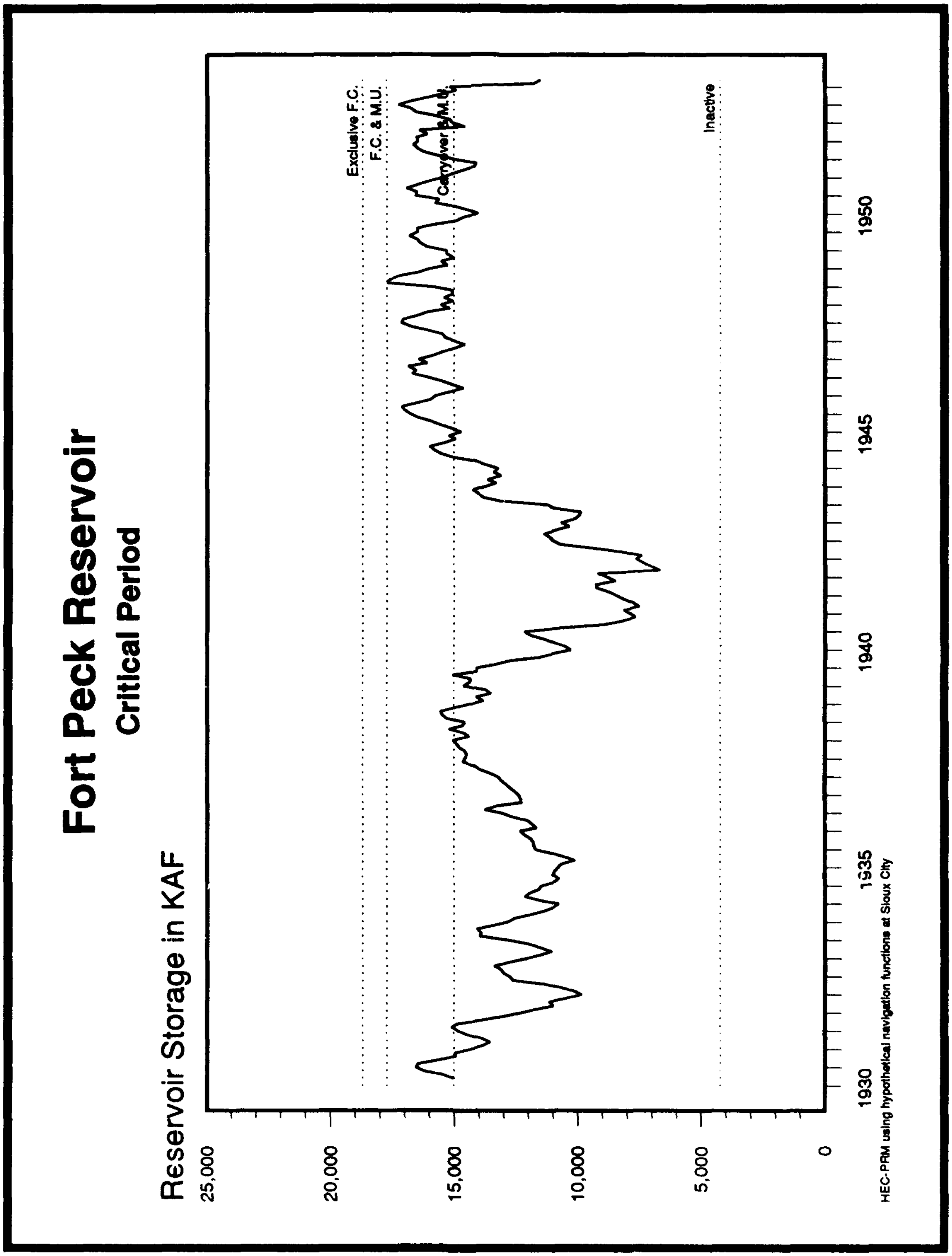


Figure 15
Ft. Peck Prescribed Storages for Critical Period with Hypothetical Navigation Penalty Function

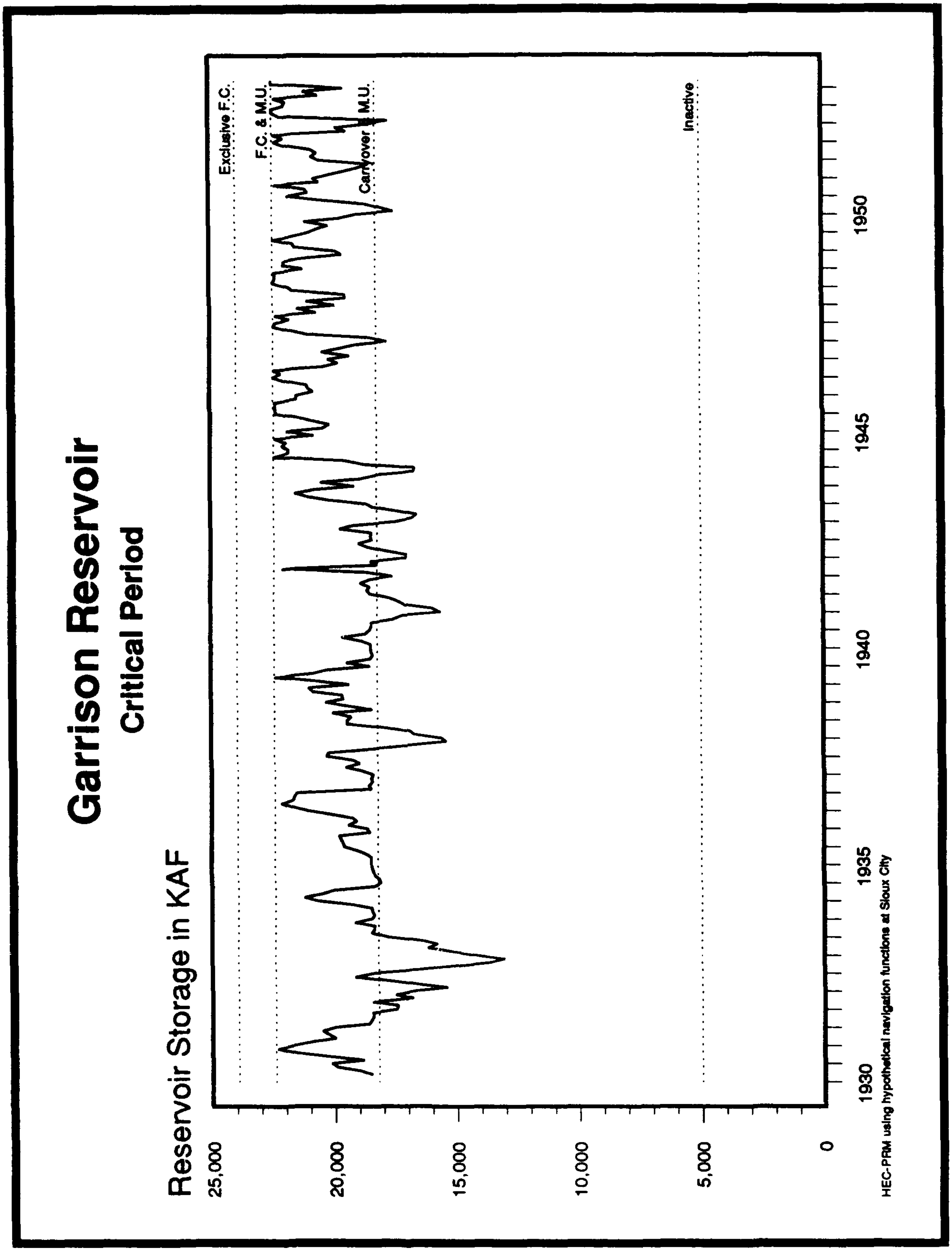


Figure 16
Garrison Prescribed Storages for Critical Period with Hypothetical Navigation Penalty Function

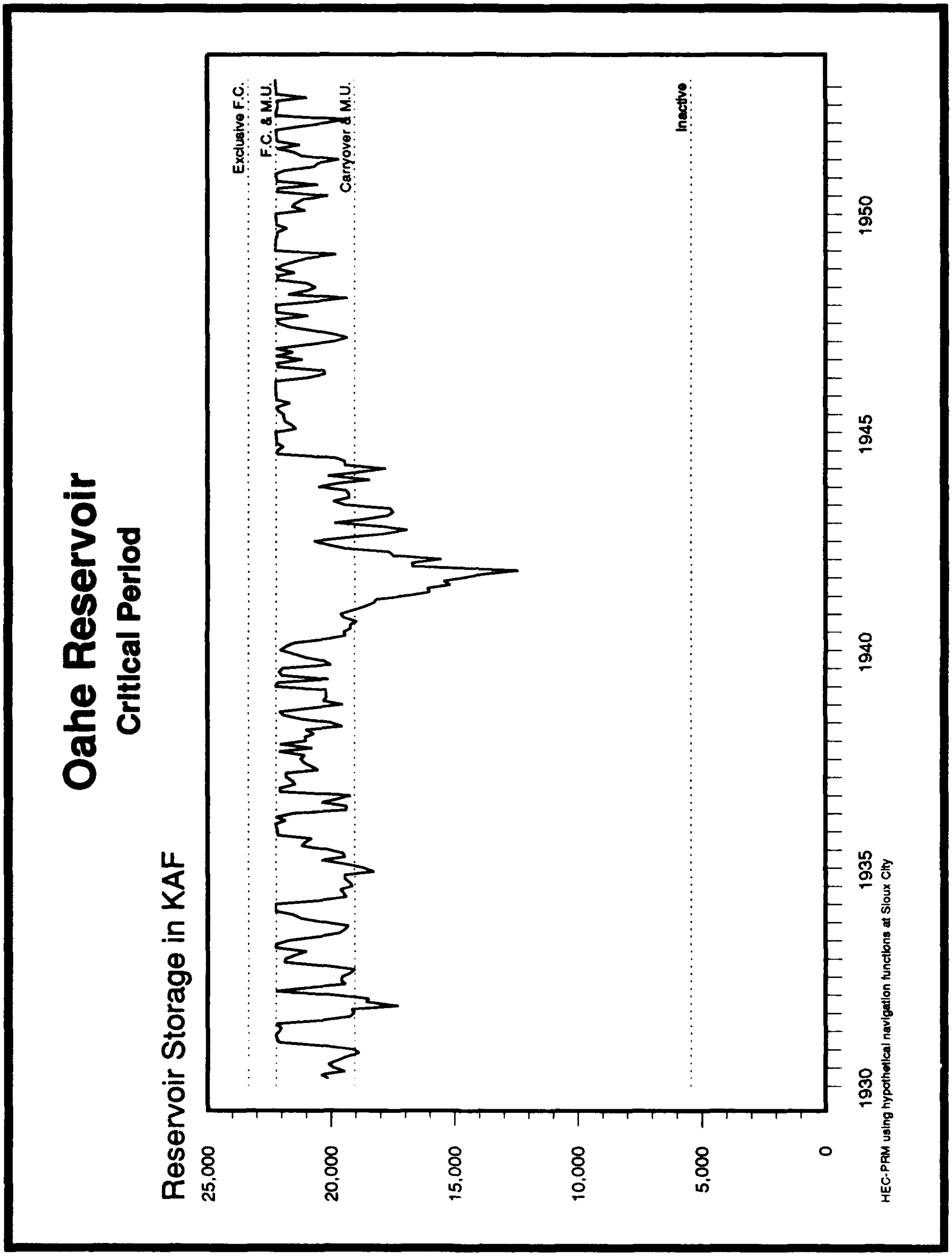


Figure 17
Oahe Prescribed Storages for Critical Period with Hypothetical Navigation Penalty Function

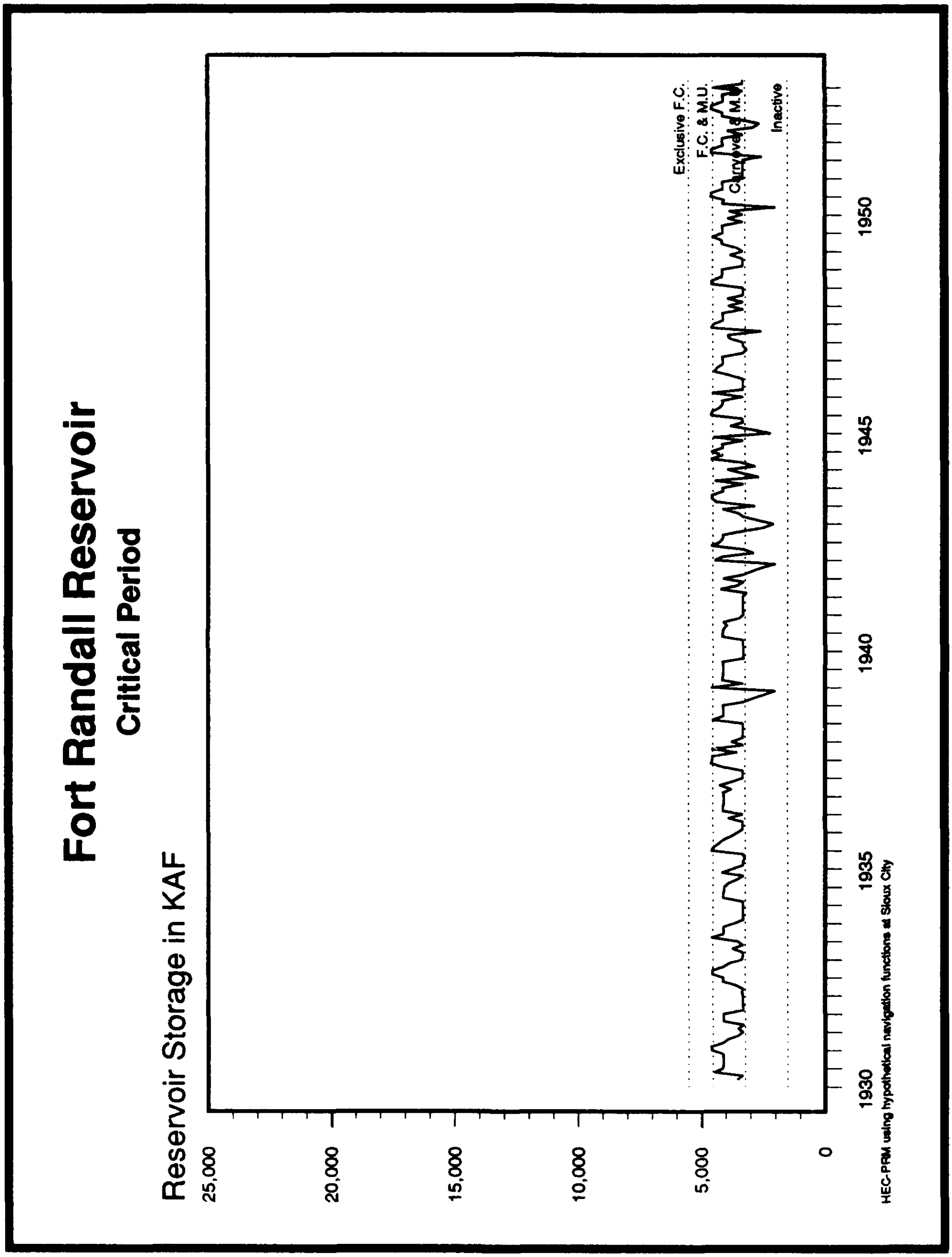


Figure 18
Ft. Randall Prescribed Storages for Critical Period with Hypothetical Navigation Penalty Function

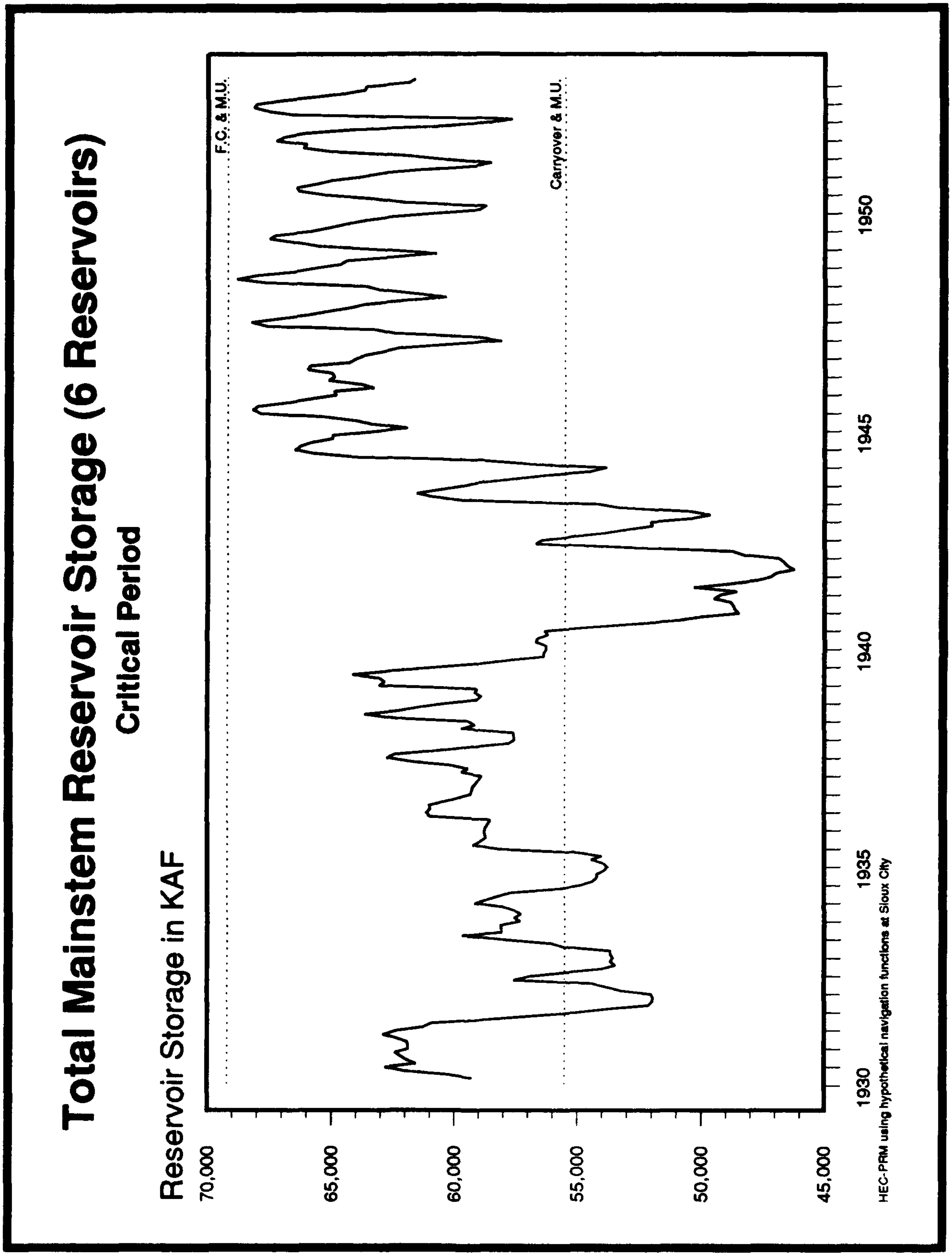


Figure 19
Total Prescribed Storages for Critical Period with Hypothetical Navigation Penalty Function

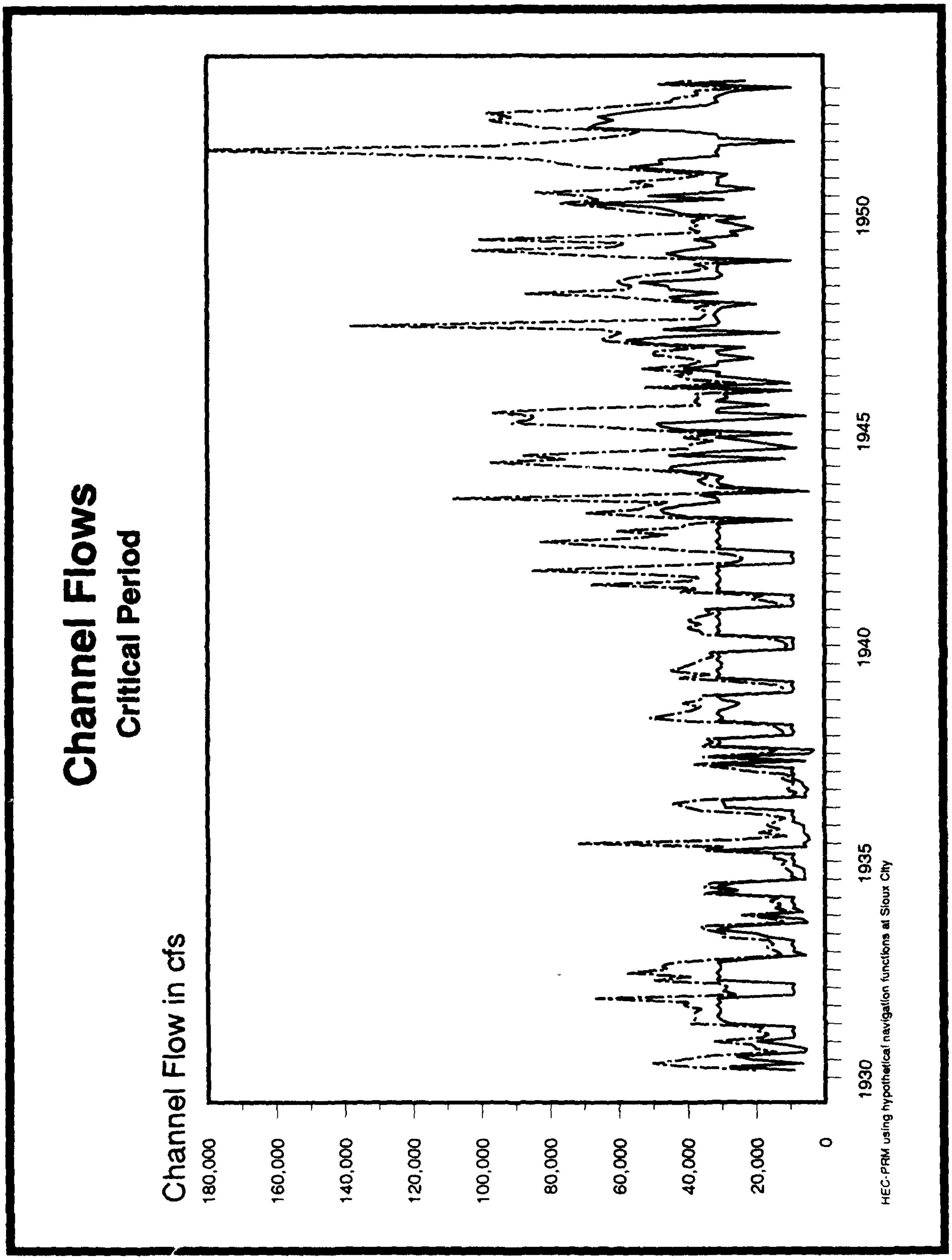


Figure 20
Sioux City and Kansas City Flows for Critical Period with Hypothetical Navigation Penalty Function

City and Kansas City in that case. Often the system has operated to provide exactly 1875 KAF/month(31,000 cfs) during April-November. For December-March, the system has reduced releases to a bare minimum to conserve water to meet subsequent April-November demands. Even so, to satisfy the 1875 KAF/month (31,000 cfs) minimum at Sioux City, the system must draw down Ft. Peck, Garrison, and Oahe, starting in 1939. For example, the January 1942 storage at Ft. Peck falls to 7000 KAF, whereas without the hypothetical function, it was approximately 15000 KAF. Earlier and later in the critical period, the Ft. Peck storages are approximately the same with and without the function. Then sufficient water is available to meet the demand without drawing on upstream storage.

Chapter 3

Phase II Activities

Summary of Activities Proposed

As proposed by HEC, Phase II of the Missouri River system study began in January 1991, with the following goals:

(1) expand the system analyzed;

(2) refine the penalty functions used;

(3) improve HEC-PRM's user interface;

(4) make technical improvements to HEC-PRM;

(5) complete additional Phase II analyses; and

(6) transfer developed technology to MRD staff.

Each of these proposals is summarized herein, and the progress is described.

Model Expansion

Proposed Expansion. Phase I HEC-PRM applications included all six main-stem reservoirs, but were limited to non-reservoir control points as far downstream as Hermann. For Phase II analyses, HEC proposed the following expansion:

(1) add St. Louis as a node to enable modeling the effects of the Missouri River reservoir operation on navigation of the Mississippi River between St. Louis and Cairo, the mouths of the Missouri and Ohio Rivers, respectively; and

(2) expand the period of analysis to the available period of record.

Accomplishments. The model was successfully expanded to the full hydrologic record: Hydrologic data files were completed for the full 1898 to 1990 presently-available record. The software permits choosing all or any selected time window for analysis. The model was successfully extended by adding a control point for St. Louis and incorporating local flow representing the Mississippi River above the junction with the Missouri River. A penalty function representing downstream Mississippi River navigation impacts was applied to flows in the St. Louis-to-Cairo reach for April through November. This function is shown as Figure 21.

Figure 22 shows total system storage prescribed by HEC-PRM, with and without the St. Louis penalty function. These results are not significantly different. This indicates that optimal operation of the system for all other purposes, ignoring navigation at St. Louis, more-or-less satisfies requirements

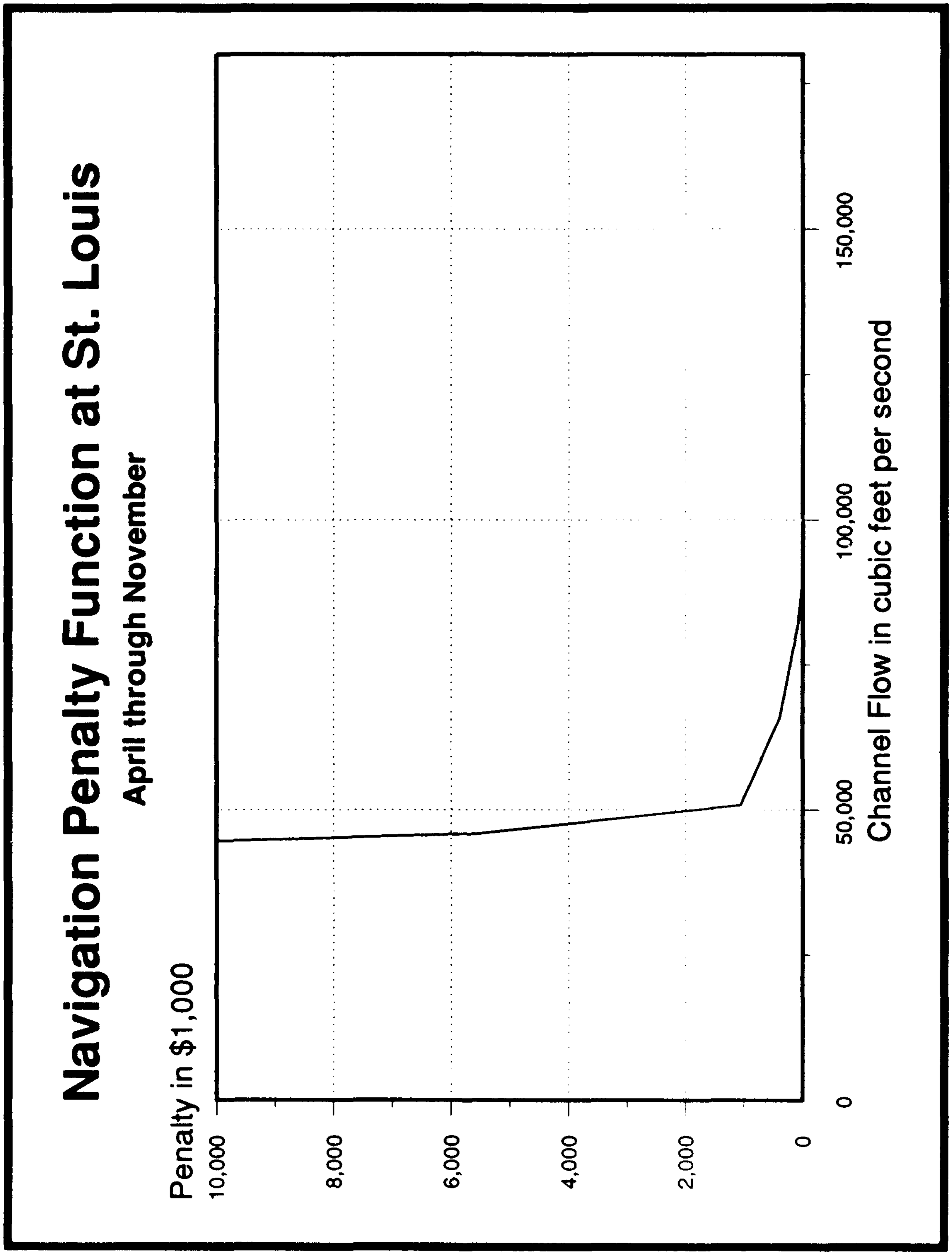


Figure 21
St. Louis Navigation Penalty Function

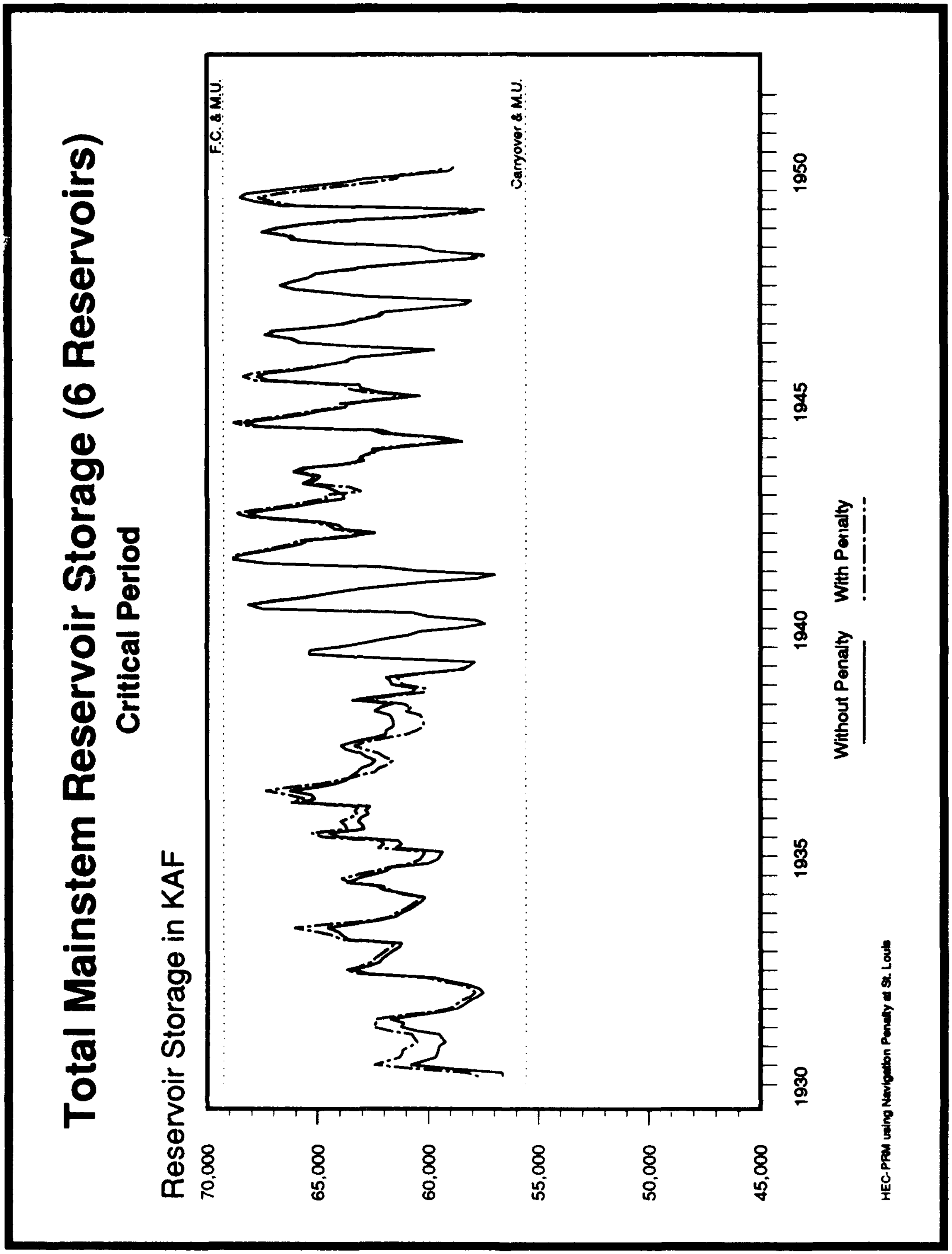


Figure 22
Total Prescribed Storage for Critical Period with St. Louis Penalty Function

for navigation flows at St. Louis. Of course, the operation for individual months varies depending on inclusion of the penalty function. However, HEC-PRM does not consider operation for individual months without reference to the long-term impact. This is illustrated well with Figure 23, flow-duration functions at St. Louis. The functions are essentially the same with and without the navigation penalty.

Penalty Function Refinement

Refinement Proposed. The penalty functions used in Phase I were based on the best currently available data. For Phase II, these functions were to be refined, and functions were to be added to permit modeling operation for all purposes. As originally proposed, this refinement was to be undertaken as a task separate from model development. In addition to function refinement, representation of short-term flood-control operation goals and objectives in the monthly analysis with HEC-PRM was to be addressed.

Accomplishments. In general, refinement of the penalty functions has lagged behind model development. Work continues by IWR and MRD staff. The penalty functions required for Phase II analyses are to be available in early 1992. As a consequence of the delay, final analyses were not completed as of the publication of this report. This is discussed further subsequently.

The flood-control issue was addressed. In Phase I, no penalty was associated with storing water in the flood-control pools of the system reservoirs. Instead, HEC-PRM was constrained to prevent use of the exclusive flood-control pool. HEC explored the following alternatives for analysis with HEC-PRM:

(1) develop penalty functions to discourage storage in the flood-control zone;

(2) continue to exclude storage from the zone; or

(3) analyze system operation with and without storage permitted in the zone.

Each alternative is described in Appendix A. After considering the pros and cons of each, HEC staff decided to continue analysis as performed in Phase I. If HEC-PRM fails to find a solution as a consequence of these constraints, the constraints can be relaxed, and the model re-run.

User Interface Improvement

Proposed Improvements. Identified targets for improvements to the HEC-PRM user interface for Phase II were:

(1) automate penalty-function derivation;

(2) improve presentation of results; and

(3) implement a viable user interface for the Phase II analyses commensurate with the time and resources available.

Accomplishments. A character-based interface was selected and implemented. This interface, designated MENUPRM, is a shell similar to that employed by HEC for its other PC-based programs.

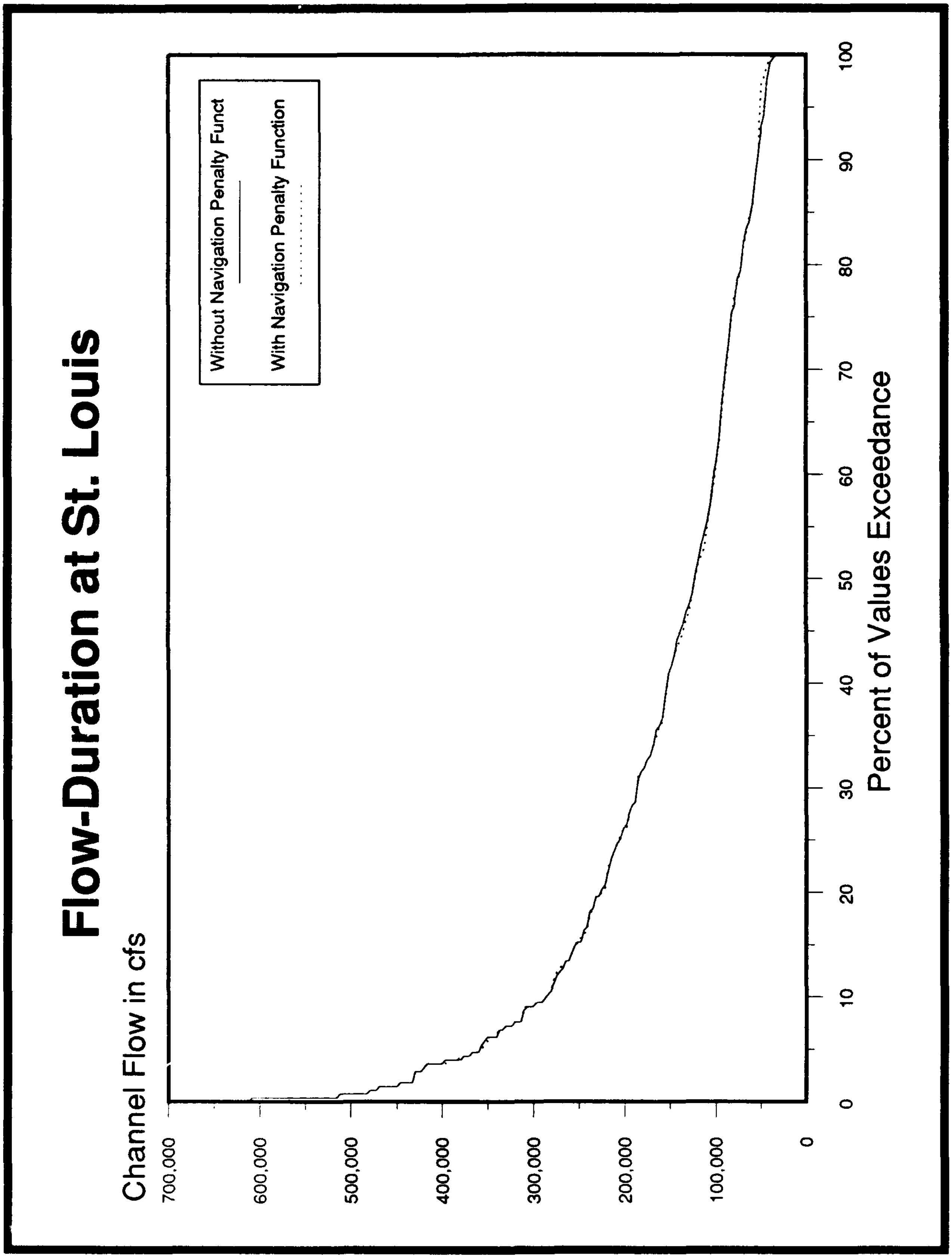


Figure 23
St. Louis Flow-Duration for Critical Period with St. Louis Navigation Penalty Function

MENUPRM provides the user easy, consistent access to HEC-PRM and the support programs required. With MENUPRM, the program user can accomplish the following:

(1) Assign a subdirectory for each study;

(2) Select a study and move to the associated subdirectory;

(3) Select the desired program from those listed and described in Appendix B;

(4) Identify files required to execute the selected program including ASCII input and output files, ASCII macro and function files, and binary files, such as HEC-DSS files;

(5) List existing files associated with the selected program;

(6) Edit files using the COED editor;

(7) Edit files using the alternate editor, "Blackbeard" from Lahey Computer Systems;

(8) Delete a file;

(9) Execute the selected program;

(10) View the ASCII output from the selected program with the LIST program;

(11) Print the files with the DOS PRINT command; and

(12) Automatically post-process results from HEC-PRM with programs PRMPOST and MATHPK.

Appendix I describes the purpose of MENUPRM and displays the various menus which are accessible to the user. It also describes the procedures and sequences of moving through the menus as well as the required key combinations for performing operations. An example operation is to execute the selected program by pressing the "Alt-X" key combination.

Results of solution of the network-flow programming representation of the reservoir operation problem are stored by HEC-PRM with the HEC-DSS. To display the results in a meaningful, consistent manner, HEC staff selected a set of standard operation reports and developed macros to produce these with program MATHPK (USACE, 1991e). MATHPK is accessible from the user interface. Appendix C describes the macros.

HEC-PRM requires the user to specify a single convex and piecewise-linear penalty function for each system link. To develop this function, the user must pre-process penalty functions developed for various project purposes. For the Phase I applications, this pre-processing was done using Lotus 123, DSPLAY, and a number of manual steps. This process was improved substantially with better interface between the Lotus 123 steps and HEC-DSS file entry. However, further improvements are desirable.

In addition to the planned activities, modifications were made to MATHPK and program PRMPOST was written to process model results. These modifications and additional programming permit convenient period-by-period comparision of HEC-PRM and L01 results.

Technical Improvements to HEC-PRM

Proposed Improvements. HEC proposed the following technical improvements to HEC-PRM for Phase II studies:

(1) the execution time required would be reduced, if feasible; and

(2) an improved algorithm would be implemented for computation of hydropower penalty.

Accomplishments. Significant progress has been made towards reducing execution time. In addition to various alternatives proposed in the Phase I report, HEC contacted and subsequently contracted with Prof. Paul Jensen of The University of Texas at Austin to improve the generalized network solver used in HEC-PRM. Jensen provided an updated FORTRAN-callable solver in May 1991 and reported solution times that were 20-50% of those required by the solver originally employed (Jensen, 1991a). Subsequently, Jensen further modified the solver to permit rapid re-optimization after changes in the model (Jensen, 1991b). This modification will be particularly useful in iteration for estimation of hydropower penalty. Unfortunately, the structure of the updated solver is significantly different from the original solver, so it has not yet been incorporated fully in HEC-PRM. However, for interim applications, software was developed to use this solver and alternative solvers with data transfer via files. With this approach, the network description is written to a file. The solver reads the file, determines the optimal flows, and writes the solution to a file. This solution file is read and processed, and the results are stored with HEC-DSS for display and further analysis.

For all applications in the Phase I study, hydropower penalty functions were simplified to express the penalty as a function of reservoir release only. For Phase II, a hydropower algorithm was developed to overcome this deficiency. This algorithm is a successive linear programming solution, based on those proposed in the following references:

Grygier, J.C., and Stedinger, J.R. (1985). "Algorithms for optimizing hydropower system operation." Water Resources Research 21(1), 1-10.

Martin, Q.W. (1982). "Multireservoir simulation and optimization model SIM-V," UM-38, Texas Department of Water Resources, Austin, TX.

Reznicek, K.K., and Simonovic, S.P. (1990). "An improved algorithm for hydropower optimization." Water Resources Research 26(2), 189-198.

The algorithm is presented in detail in Appendix D. In summary, the approach is to estimate storage (head) for each reservoir, solve the resulting linear problem in terms of release, re-compute storage given the optimal release, and re-iterate if the difference between estimated and computed storage exceedes a tolerance. Initial tests of the algorithm were made. Final coding of the algorithm and testing is not yet complete, pen. .ıg incorporation of the improved network solver.

Phase II Analysis

Analyses Proposed. For Phase II, HEC proposed to perform three or four analyses of system operation, in anticipation that MRD staff would request these for their studies. HEC staff expected that these analyses would represent differing views regarding system operation that would surface during model development. HEC planned that these analyses would use complete, finalized penalty functions with a final, Phase II version of HEC-PRM.

Analyses Completed. An MRD request was not received, and the proposed modeling was not performed. The delay occurred because of unanticipated delays in development of complete penalty functions

Technology Transfer

Proposed Transfer Activities. HEC proposed the following: (1) prepare a draft user's manual; and (2) conduct a workshop on model application for MRD staff.

Accomplishments. A successful workshop was conducted in Davis, CA, on 2-4 December 1991 for MRD staff. The agenda for that course is included as Appendix E. Additional workshop materials not included in this report are published under separate cover.

The following additional documents were prepared to provide MRD staff with guidance in use of HEC-PRM and are included as Appendices to this report:

(1) *HEC-PRM Pathname Parts, MRD Application. (Appendix G)*

(2) *HEC-PRM Program Description. (Appendix F)*

(3) *HEC-PRM Supplemental Programs. (Appendix H)*

In addition to these manuals, two internal reports were prepared to documen improvements and modifications to the network solver (Jensen, 1991a,b). Another report is under preparation to provide guidance in developing operating plans from HEC-PRM results. This report is not scheduled for completion until after publication of this Phase II report. Preliminary proceedings and findings of this report were presented in the December 2-4 workshop and are summarized in the next chapter

Chapter 4

Development of Operation Plans from Analysis Results

As an extension of the Phase II activities, the Hydrologic Engineering Center is reviewing alternatives to enable recommendation of procedures for developing system operating rules from HEC-PRM results. This task is necessary to bridge the gap between theory (the prescriptive model) and practice (the real-world system). The present status and findings of this activity were presented in the 2-4 December 1991 workshop and will be discussed in a future technical report by Prof. Jay Lund, contractor to the HEC (USACE, 1992). These are summarized briefly herein.

Some Uses of Operations Plans with HEC-PRM Results

There are two potential uses of operating rules developed from HEC-PRM results for the updating of reservoir control manuals. The first use is to suggest modifications in the rules currently used to operate a reservoir system. Here, the HEC-PRM results are used to "re-calibrate" existing operation rules, varying parameters such as minimum pool sizes, the rate at which releases increase with storage, etc. The second potential use of rules developed from HEC-PRM results is to suggest new, quantitatively different operating rules for the system. This second use can be used to suggest near-optimal operating procedures that may be very different from the approach currently used. This provides a relatively rigorously-derived alternative to modifications to existing operating rules. These alternatives can then be compared on the basis of more detailed simulation studies.

Approaches to Developing Operations Plans with HEC-PRM Results

There are several general approaches to developing operations plans from optimization model results. These include intuitive approaches, use of statistical regression, the use of balancing rules from reservoir operations theory, and mixed simulation-optimization approaches. These are summarized in Table 4.

In actual practice, a combination of all these approaches, in some mix, has typically been used in the literature. In any approach taken to develop operation plans from optimization results, the operation plan suggested by the optimization results should only be viewed as an initial suggestion for the reservoir system's operations. The nature of optimization modeling typically requires simplification of the problem to a significant degree beyond the simplifications made in most system simulation models. Therefore, any optimization-derived operation plan should be considered subject to further refinement and testing with the aid of simulation models. In this way, the operation plan from the optimization results is mainly a "point of departure".

Approach Taken for MRD Application

The HEC-PRM results used in developing preliminary operation rules utilized the full 92-year hydrologic record for the basin. This long record contained several major drought and flood events and was assumed to be representative of the range and frequency of conditions that will be experienced in the future.

Table 4
Alternative Approaches for Developing Operation Rules from HEC-PRM Results

Approach	Description
Intuitive	This approach relieves on the analyst's subjective review of HEC-PRM results to identify patterns in operation. For example, if HEC-PRM consistently prescribes seasonal drawdown of a given reservoir, the analyst could infer that a good operating rule for that reservoir would include such seasonal drawdown.
Regression	This approach was formally proposed by G.K. Young in 1966 in his paper "Finding reservoir operating rules". The approach relies on statistical analysis to identify any correlation between the optimal releases, as defined by a prescriptive model, and system inputs and state variables. For example, a regression equation might be developed to define the optimal monthly reservoir release as a function of beginning-of-month storage and predicted monthly inflow.
Balancing rules	This approach seeks to identify patterns in the prescriptive-model results that are consistent with theoretical operation rules. These theoretical rules include, for example, the space rule, the pack rule, and the hedging rule described by Maas, et al.(1962).
Mixed simulation	Although identified as a separate optimization approach, Lund characterizes this as a critical component of all approaches. He proposes that any operating rules or patterns identified be tested with detailed simulation model. This will permit more detailed review of both the prescriptive model results and the derived operation rules.

In analyzing the record, an "informal, yet systematic" strategy is used to infer new operating rules from "optimal" HEC-PRM results. This strategy incorporates some elements of all the approaches mentioned above and employed a statistical package to quickly display and process results data. First, the HEC-PRM results were reviewed graphically with time series plots and histograms and with the use of descriptive statistics, mean monthly storages, minimum storages, mean releases, etc. This identified consistent and dominant trends in the HEC-PRM results. This first pass was able to identify near optimal storage levels for each of the lower three reservoirs for each month and identified minimum storages in all of the system's reservoirs. These results were very consistent across the 92-year record. This first pass fixed the annual and over-year operating behavior of the three lower reservoirs.

A second pass was then made to examine specifically the operation of the upper three reservoirs. While distinct minimum storages were evidenced for each of these three reservoirs, no simple pattern of storage or release targets was evidenced in the first pass through the results. The second pass consisted of the use of several trial regression equations to explain the behavior of the storages and releases from these reservoirs. These regressions proved largely ineffective. A second, simpler approach was then taken, plotting the storage in each reservoir against total system storage. This yielded rather consistent patterns of storage allocation between the three reservoirs as shown in Figure 24. This second pass then produced allocation rules for total storage between the three large upper reservoirs.

A third pass was then made to estimate monthly release rules from Gavins Point, based on total system storage and inflows. With this estimate, total storage can be cumulatively derived and the

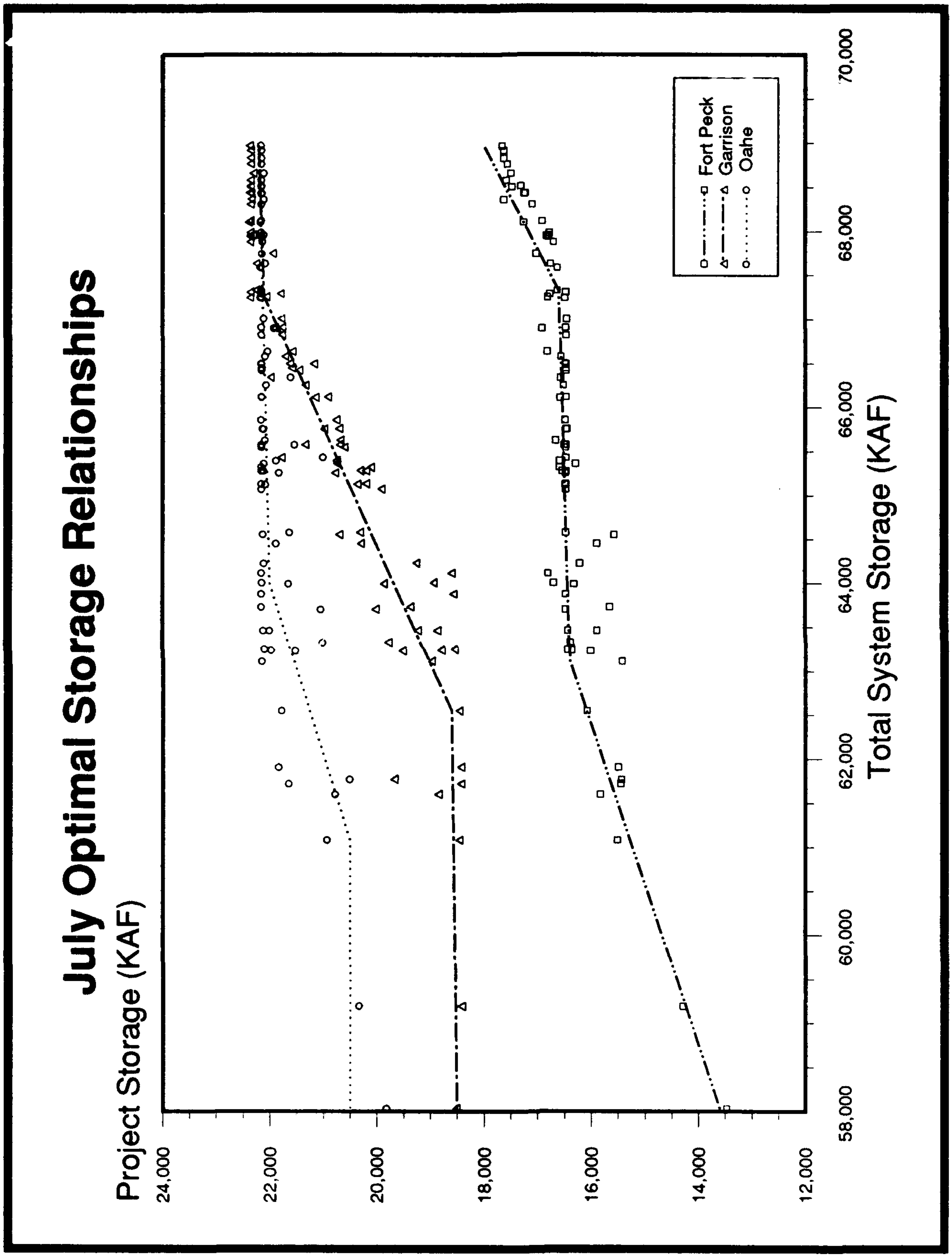


Figure 24
July Optimal Storage Relationships

operation of the entire system is then specified.

Finally, a comparison of the operations suggested by HEC-PRM and the current operation plan, represented by the MRD system simulation model, was made. This pointed out divergences between HEC-PRM's suggested operations and current operation plans. Had this been an actual application to actually suggest new operation plans, a series of follow-up simulation studies would be required.

The process of finding operating rules from the 92-year HEC-PRM results revealed the following causes for concern, which should be addressed before final HEC-PRM results are created:

1. It appears that the preliminary penalty functions for flooding in the winter are the same as those for summer months. This neglects the important role of ice formation on winter flooding discussed in the current water control manual. The absence of ice-related flooding penalties drastically changes the within-year operation of the system, retaining within-year storage much later in the winter. The penalty functions for flooding due to ice formation are currently under development and will be used in the final HEC-PRM runs.

2. In comparing the HEC-PRM results with the results from the MRD simulation model, it is evident that the same hydrologic input was not used as input to the two models. To make accurate comparisons, the proper versions of depletions and models must be used so that both programs are analyzing the same level of depletions and that the input is of the same version as that of the model.

Final work with HEC-PRM to suggest operation plans for the main stem system cannot be completed until final penalty functions are derived. When these become available, the final 92-year runs could be made and interpreted for the development of operation plans. From the interpretation of the initial 92-year run, it appears that this would be possible. However, if one were to develop final operation plans from HEC-PRM results, the functions would have to be refined and tested by simulation.

Chapter 5

Planned Model Improvements and Their Impact on Future Analysis

In calendar year 1992, HEC staff plan improvements to HEC-PRM for other on-going studies. These improvements include, but are not limited to, the following:

(1) further refinement of the user interface;

(2) integration and testing of the improved network solver; and

(3) implementation and testing of the hydropower algorithm.

The goal is to develop a general-purpose model and software that can be applied on any system. The current HEC target for public release of the software is 1 January 1993. However, as HEC-PRM is improved in the interim, updates will be provided to MRD.

User Interface Refinement

The user interface described in Chapter 3 is viewed as an evolving product. Changes to it will make HEC-PRM easier and more convenient to use by enhancing input data preparation and streamlining the review of results. Additional capabilities anticipated include improvements to the method of defining composite penalty functions. These improvements will permit users to analyze operation options more efficiently in response to proposals by those affected by system operation. No major change in the overall structure of the analysis package (HEC-PRM coupled with HEC-DSS) is anticipated. Consequently, changes to the user interface will be "upwardly compatible," permitting continued use of previously-developed time series data, penalty functions, and HEC-DSS input data files.

Improved Network Solver

The existing HEC-PRM network solver will be replaced with the more-efficient solver provided by Jensen and described in Chapter 3. This should reduce the computation time required for analysis of any proposed operation goals and objectives. To incorporate the new solver, the FORTRAN code of HEC-PRM will be restructured as necessary. This restructuring will yield code that is oriented towards the new solver's theoretical methodology (primal simplex) as opposed to the old solver's theoretical methodology (dual flow augmentation). This will facilitate the efficient replacement of Jensen's solver with a even more efficient commercial solver of the same theoretical methodology if it is deemed necessary.

In addition to the anticipated reduction in execution time due to the improved solver, a reduction in memory requirement is anticipated. As described in earlier documents (USACE, 1991b), the current version of HEC-PRM represents nonlinear penalty functions with a piecewise linear approximation. Each segment of the approximation requires a separate arc of the network. The improved solver uses a new data structure for storing models with piecewise linear arc costs. This structure requires fewer arcs to represent the same system. This reduction in memory is offset by the new solver's requirement of double precision variables for some solver matrix variables. If it results in a net reduction of memory requirements, this would enable users to expand the system analyzed, include more control points, or optimize for a longer period of time.

Hydropower Algoritnm

The hydropower algorithm described in Chapter 3 will be implemented and tested, following implementation of the improved solver. This algorithm will permit more accurate analysis of system operation for energy generation, as it will evaluate properly the penalty as a nonlinear function of storage and release. In the case of the Missouri system, this will yield a "fine tuning" of simulation results.

References

Chung, F.I., Archer, M.C., and DeVries, J.J. (1989). "Network flow algorithm applied to California aqueduct simulation," *Journal of the WRPMD*, ASCE, 115(2), 131-147.

Ford, L.R., and Fulkerson, D.R. (1962). *Flow in Networks.* Princeton University Press, Princeton, NJ.

Ford, D.T., and Davis, D.W., (1989). "Hardware-store rules for system-analysis applications," in *Closing the Gap Between Theory and Practice*, ed. D.P. Loucks, Baltimore, MD.

Grygier, J.C., and Stedinger, J.R. (1985). "Algorithms for optimizing hydropower system operation." *Water Resources Research* 21(1), 1-10.

Ikura, Y., and Gross, G. (1984). "Efficient large-scale hydro system scheduling with forced spill conditions," *IEEE Transactions on Power Apparatus and Systems*, PAS-103(12), 3501-3520.

Jensen, P.A., Bhaumik, G., and Driscoll, B. (1974). "Network flow modeling of multireservoir water distribution systems," *CRWR-107*, University of Texas, Austin, TX.

Jensen, P.A., and Barnes,J.W. (1980). *Network Flow Programming.* John Wiley and Sons,

Jensen, P.A., (1991a). "Development of a minimum cost network-flow solver," Report to the Hydrologic Engineering Center, Davis, CA.

Jensen, P.A., (1991b). "Enhancements to the minimum cost network-flow solver," Report to the Hydrologic Engineering Center, Davis, CA.

Liebman, J.C. (1976). "Some simple-minded observations on the role of optimization in public systems decision making," *Interfaces*, TIMS, 6(4), 102-108.

Maas, A., et al. (1962). *Design of Water-resource Systems.* Harvard University Press, Cambridge, MA.

Sabet, M.H., Coe, J.Q., Ramirez, H.M., and Ford, D.T. (1985). "Optimal operation of California aqueduct," *Journal of the WRPMD*, ASCE, 111(2), 222-237.

Texas Water Development Board (1974). "Analytical techniques for planning complex water resource systems," *Report 183.* Austin, TX.

USACE (1979). *Missouri River Main Stem Reservoir System Reservoir Regulation Manual: Master Manual.* U.S. Army Engineer Division, Missouri River, Omaha, NE.

USACE (1990a). *Plan of Study for the Review and Update of the Missouri River Main Stem Reservoir System Reservoir Regulation Manual.* U.S. Army Engineer Division, Missouri River, Omaha, NE.

USACE (1990b). *Economic Value Functions for Missouri River System Analysis Model - Phase I.* Institute for Water Resources, Ft. Belvoir, VA.

USACE (1990c). *HEC-DSS Users's Guide and Utility Program Manuals.* Hydrologic Engineering Center, Davis, CA.

USACE (1991a). *HEC-PRM Pathname Parts, MRD Application.* Hydrologic Engineering Center, Davis, CA.

USACE (1991b). *Missouri River System Analysis Model - Phase I.* Hydrologic Engineering Center, Davis, CA.

USACE (1991c). *HEC-PRM Program Description.* Hydrologic Engineering Center, Davis, CA.

USACE (1991d). *HEC-PRM Supplemental Programs.* Hydrologic Engineering Center, Davis, CA.

USACE (1991e). *MATHPK: Mathematical Manipulation of Data Stored in a DSS Data File, Users Manual.* Hydrologic Engineering Center, Davis, CA.

USACE (1992). *Developing Operation Plans from HEC Prescriptive Reservoir Model Results for the Missouri River System: Preliminary Results.* Hydrologic Engineering Center, Davis, CA.

USACE (undated). "EDP program: Missouri river main stem reservoirs - long range regulation studies," *MRD-RCC Technical Report F-81.* Omaha, NE.

Yeh, W. W-G. (1985). "Reservoir management and operations models: A state-of-the-art review," *Water Resources Research*, 21(12), 1797-1818.

Young, G.K. (1966). "Finding reservoir operating rules," *Journal of the Hydraulics Division*, ASCE, 93(HY6), 297-319.

Appendix A

Alternatives for Modeling Flood-Control Operation

Alternatives for Modeling Flood-Control Operation

Penalty Functions

Description. Penalty functions can be developed to discourage use of reservoir flood-control storage. These functions would represent as a function of beginning of month storage, for example, the short-term (within-month) risk of storing rather than releasing water. These penalties can be compared with penalties of releasing water, thus evaluating trade-offs in operation.

Single Reservoir or Independent Operation. For a single reservoir or for a system in which reservoirs are operated independently for flood control, a flood-control storage penalty function can be developed as follows:

a. Develop a set of hypothetical inflow hydrographs, each with specified frequency. These hydrographs represent short-term flood inflow. Route these hydrographs to downstream locations to define unregulated frequency curves. Compute the damage associated with each peak. Define a damage-frequency function. Integrate to compute unregulated expected damage.

b. For a selected beginning-of-month (BOM) state of flood-control storage (for example, flood-control pool empty), determine the downstream damage with each hydrograph if the reservoir is operated according to existing rules. Determine the corresponding damage. Develop a damage-probability function. Compute expected damage. Compare unregulated and regulated damage to compute expected benefit. Assign the benefit to the corresponding BOM storage state.

c. Repeat step b for other selected BOM storage states to develop a complete function.

This procedure is described in some detail in the following:

Duren, F.K., and Beard, L.R. (1972). "Optimizing flood control allocation for a multipurpose reservoir." *Water Resources Bulletin*, 8(4), 735-744.

Multiple Reservoir System. Developing flood-control storage functions for a multiple reservoir system is more complex. In that case, the damage prevented by one reservoir (and hence, the value of empty storage in that reservoir) depends on the available storage in other reservoirs and on the inflows to those reservoirs. Such nonseparable functions cannot be represented conveniently with a linear model such as HEC-PRM. Possible remedies to this problem include the following:

a. Assume all reservoirs in same condition of storage (ie., if one reservoir flood-control pool is empty, all flood-control pools are empty). Assume probability of inflows same at all sites (ie., all reservoirs have n-year flood inflow simultaneously). Perform computations as above.

b. Assume state of storage and system inflows are independent. Select a BOM storage state for one reservoir. Generate a large set of random inflows for system reservoirs, with all other reservoir BOM storage states selected randomly. Evaluate average damage prevented and assign to the selected BOM storage state of the reservoir in question. Repeat for other BOM storage states. Then repeat for all other reservoirs.

Pros and Cons. This alternative makes a legitimate attempt to value storage for trade-off analysis. Unfortunately, the methods are not well documented or understood. Development of the functions as proposed herein is computationally intensive.

Exclude Storage

Description. In phase I of the Missouri River system, HEC included flow-related penalty functions to represent damage at downstream control points. After early runs revealed that the model called for storing significant volumes in the exclusive flood-control pool, HEC added inviolable constraints to prohibit use of that storage. In that case, HEC-PRM prescribes only flows and storages that do not require use of the flood-control space. If HEC-PRM could not allocate water in some fashion that did not require use of the excluded storage, it would declare that no solution exists.

Pros and Cons. This alternative is simple. However, it makes no attempt to value storage for trade-off analysis. Adding constraints to HEC-PRM assumes that the existing allocation of storage is inviolable. If the flood-control space is required to manage a large, long duration flood, no feasible solution will be possible.

Analysis With and Without Storage Permitted

Description. With and without analysis with HEC-PRM permits the analyst to impute the value of various operation constraints. For example, the analyst could add constraints to prohibit use of flood-control space in Ft. Peck and re-analyze system operation. The analyst could then eliminate the constraint and re-analyze operation. The difference in system penalty is the value of permitting use of the flood-control space. Many variations on this theme are possible, including use of a portion of the space, use of space in various combinations of reservoirs, use in some months only, etc.

Pros and Cons. This alternative also is simple to use. It does make a legitimate attempt to value storage for trade-off analysis. However, it has the same problems as excluding storage if storage is required.

Appendix B

Computer Programs Accessable from MENUPRM

Computer Programs Accessable from MENUPRM

Program	**Description**
HEC-PRM	This is the prescriptive model. This program formulates and solves the network-flow programming problem that represents the reservoir operation problem. The network-flow solver is integrated into this program.
PRMPOST	This program post-processes the HEC-PRM results, yielding a table similar in format to the output of L01.
DSSUTL	This is the HEC-DSS housekeeping program. It permits the user to tabulate, edit, copy, rename, and delete data stored with the HEC-DSS. It also converts data to and from ASCII format from HEC-DSS format.
DSPLAY	This is the graphing program for data stored with the HEC-DSS. It will graph both time series and pair-function data.
MATHPK	This program is the equivalent of a spreadsheet for data stored with the HEC-DSS. It provides extensive capabilities to perform computations with stored data.
DSSPD	This program allows the user to store paired-function data with the HEC-DSS. It may be used to enter penalty functions directly for analysis with HEC-PRM.
RDMATF	This program reads the output file created by MRD's post-processor, V1.EXE, and stores the results in an HEC-DSS file.
RDATA0	This program read's MRD's file DATA0 and stores the data in an HEC-DSS file.
DSSTXT	With this program, the user can store or retrieve text from an HEC-DSS file.
DSSTS	With this program, the user can enter regular-interval time series into a HEC-DSS file from keyboard or a file.
L01	This is MRD's long-range-study model. It simulates operation of the six major reservoirs of the Missouri system for six downstream control points for 15 periods per year.
V1	This is the post-processor for L01. It provides the user with a menu of system variables computed for various locations. The user can tabulate or plot these values.
COED	This is the Corps editor. It permits the user to modify any text file. It is used to prepare and/or edit the HEC-PRM input file. Through file conversions, it also may to used to edit data stored with HEC-DSS.

LIST This program permits the user to "browse" an ASCII file on-screen.

Appendix C

MATHPK Macros for HEC-PRM

MATHPK Macros for HEC-PRM

Listing of Macros

Description of MATHPK Macros

DO_PRM	Master macro to process all reservoirs and control points. If reservoirs are added or removed, this macro must be modified. There is one line for every control point and 3 lines for every reservoir. The reservoir lines are (1) for most information including elevation, flow in both units, time series penalties, etc., (2) for hydropower energy, and (3) for hydropower capacity.
DO_PRM_M	Macro similiar to DO_PRM. This acts as a backup.
DO_PRM1	Macro similar to DO_PRM except this is for a test case of one reservoir and one control point.
DO_MRD	Simpler form of DO_PRM for only some information.
OPEN1	Opens all DSS data files. It assumes that MRD.DSS is the time series file and PENCMP.DSS is the penalty function file. It calls the more complex macro OPEN.
OPEN	Opens all DSS data files, sets the time window, sets the interpolation paramater to no extrapolation, sets the number of lines of printout to 46, initializes the tabulation file so that all tabulations will start at the beginning of the file, gets certain parameters like the month number, the factor to convert cfs to KAF, and sets the printout format for the month number.
GET_TS	Allows a simple retrieval of time series data and the conversion from KAF to cfs.
MONVARY	Calculates time series penalties for a parameter when the penalty functions for all months are stored in one record (pathname). The penalty function record contains 12 curves, one for each month.
MONVARY1	Calculates time series penalties for a parameter when the penalty function for each month is stored in a separate record (pathname). To compute penalties for all 12 months of the year, 12 separate functions must be retrieved.
MONSAME	Calculates time series penalties for a parameter when there is only one penalty function and it applies to all months of the year.
CLASSDEF	Computes class intervals for the user given a starting value, ending value, and an interval. The user must set the pathname parts and units and then store the class intervals in the DSS data file.
COUNTLO	Computes the number of monthly occurances within a year for each year. The number of occurances is the count of the number of times a parameter falls below a certain value. Typically used for computing

the number of months during the year in which a navigation level was not met.

RES_STAT	Computes statistics associated with reservoirs and sets the the units and curve labels for the statistics variable. This macro calls STAT1.
CP_STAT	Computes statistics associated with control points and sets the units and curve labels for the statistics variable. This macro calls STAT1.
STAT1	Computes statistics of an input variable. The statistics include the minimum, maximum, mean, standard deviation, and the sum.
DO_RES	Primary reservoir macro. It calls the macros DO_RES1, RES_STAT, DO_RES2, DO_RES3, and RES_DUR. Those macros include getting storage and flow data, computed time series penalties, computing statistics, tabulating the results, storing the results in the output DSS data file, and computing duration statistics.
DO_RES1	Gets storage, flows, and the elevation-area-capacity curve. Computes pool elevation, flow in cfs, and time series penalties.
DO_RES2	Sets formats for variables defined in macro DO_RES1 and RES_STAT, and tabulates the results to the tabulation file defined in the MENUPRM.
DO_RES3	Sets units and pathname parts for variables defined in macro DO_RES1, and stores the data in the output DSS data file.
DO_CP	Primary control point macro. It calls the macros DO_CP1, CP_STAT, DO_CP2, DO_CP3, and CP_DUR. Those macros include getting flow data, computing time series penalties, computing statistics, tabulating the results, storing the results in the output DSS data file, and computing duration statistics.
DO_CP1	Gets flows and converts them to cfs. Computes time series penalties.
DO_CP2	Sets formats for variables defined in macro DO_CP1 and CP_STAT, and tabulates the results to the tabulation file defined in the MENUPRM.
DO_CP3	Sets units and pathname parts for variables defined in macro DO_CP1, and stores the data in the output DSS data file.
SYSTEM	Computes system paramaters above and including Sioux City. This macro is not complete as it does not include calculations of hydropower.
RES_DUR	Computes duration parameters at reservoirs. Includes values for reservoir pool elevation and reservoir release. Calls macro DURATION.

CP_DUR	Computes duration parameters at control points for flow in KAF. Calls macro DURATION.
DURATION	Performs duration analysis for some parameter. It includes the number of hits within each class interval, the accumulated number of hits both in increasing and decreasing value direction, and computes the percent values which equal or exceed a given class interval.
POWER	Computes hydropower energy and capacity at all reservoirs except Gavins Point.
POWER_P	Stores hydropower variables defined in macro POWER in output DSS data file.
POWER1	Computes the hydropower energy and capacity at Gavins Point.
POWER1_P	Stores hydropower variables defined in macro POWER1 in the output DSS data file.
FLOW_LOC	Computes adjusted local incremental inflows at all control points and reservoirs. It calls macro Q_LOC1.
Q_LOC1	Computes the adjusted local incremental inflows at all nodes using the unadjusted inflow, the time varying depletions, and the monthly varying depletions.

```
MACRO DO_PRM prm_dssfn pf_dssfn mpk_dssfn ts_dssfn alternative start_date end_date
!RUN OPEN prm_dssfn pf_dssfn mpk_dssfn ts_dssfn start_date end_date

!RUN DO_RES FTPK FTPK-GARR FTPK-FTPKR FTPKR-GARR 'alternative'
!RUN POWER FTPK FTPK FTPK-FTPKR 'alternative'
!RUN POWER_P FTPK FTPK FTPK-FTPKR 'alternative'

!RUN DO_RES GARR GARR-OAHE GARR-GARRR GARRR-OAHE 'alternative'
!RUN POWER GARR GARR GARR-GARRR 'alternative'
!RUN POWER_P GARR GARR GARR-GARRR 'alternative'

!RUN DO_RES OAHE OAHE-BEND OAHE-OAHER OAHER-BEND 'alternative'
!RUN POWER OAHE OAHE OAHE-OAHER 'alternative'
!RUN POWER_P OAHE OAHE OAHE-OAHER 'alternative'

!RUN DO_RES BEND BEND-FTRA BEND-BENDR BENDR-FTRA 'alternative'
!RUN POWER FTRA BEND BEND-BENDR 'alternative'
!RUN POWER_P FTRA BEND BEND-BENDR 'alternative'

!RUN DO_RES FTRA FTRA-GAPT FTRA-FTRAR FTRAR-GAPT 'alternative'
!RUN POWER FTRA FTRA FTRA-FTRAR 'alternative'
!RUN POWER_P FTRA FTRA FTRA-FTRAR 'alternative'

!RUN DO_RES GAPT GAPT-SUX  GAPT-GAPTR GAPTR-SUX  'alternative'
!RUN POWER1 GAPT GAPT-GAPTR 'alternative'
!RUN POWER1_P GAPT GAPT-GAPTR 'alternative'

!RUN DO_CP SUX-OMA SUX-OMA 'alternative'
!RUN DO_CP OMA-NCNE OMA-NCNE 'alternative'
!RUN DO_CP NCNE-MKC NCNE-MKC 'alternative'
!RUN DO_CP MKC-BNMO MKC-BNMO 'alternative'
!RUN DO_CP BNMO-HEMO BNMO-HEMO 'alternative'
!RUN DO_CP HEMO-STL HEMO-STL 'alternative'
..      !RUN DO_CP STL-S_SINK STL-S_SINK 'alternative'

!RUN SYSTEM 'alternative'

FI
ENDMACRO
```

```
MACRO DO_PRM_M prm_dssfn pf_dssfn mpk_dssfn ts_dssfn alternative start_date end_date
!RUN OPEN prm_dssfn pf_dssfn mpk_dssfn ts_dssfn start_date end_date

!RUN DO_RES FTPK FTPK-GARR FTPK-FTPKR FTPKR-GARR 'alternative'
!RUN POWER FTPK FTPK FTPK-FTPKR 'alternative'
!RUN POWER_P FTPK FTPK FTPK-FTPKR 'alternative'

!RUN DO_RES GARR GARR-OAHE GARR-OAHER OAHER-OAHE 'alternative'
!RUN POWER GARR GARR GARR-GARRR 'alternative'
!RUN POWER_P GARR GARR GARR-GARRR 'alternative'

!RUN DO_RES OAHE OAHE-BEND OAHE-OAHER OAHER-BEND 'alternative'
!RUN POWER OAHE OAHE OAHE-OAHER 'alternative'
!RUN POWER_P OAHE OAHE OAHE-OAHER 'alternative'

!RUN DO_RES BEND BEND-FTRA BEND-BENDR BENDR-FTRA 'alternative'
!RUN POWER FTRA BEND BEND-BENDR 'alternative'
!RUN POWER_P FTRA BEND BEND-BENDR 'alternative'

!RUN DO_RES FTRA FTRA-GAPT FTRA-FTRAR FTRAR-GAPT 'alternative'
!RUN POWER FTRA FTRA FTRA-FTRAR 'alternative'
!RUN POWER_P FTRA FTRA FTRA-FTRAR 'alternative'

!RUN DO_RES GAPT GAPT-SUX  GAPT-GAPTR GAPTR-SUX  'alternative'
!RUN POWER1 GAPT GAPT-GAPTR 'alternvative'
!RUN POWER1_P GAPT GAPT-GAPTR 'alternative'

!RUN DO_CP SUX-OMA SUX-OMA 'alternative'
!RUN DO_CP OMA-NCNE OMA-NCNE 'alternative'
!RUN DO_CP NCNE-MKC NCNE-MKC 'alternative'
!RUN DO_CP MKC-BNMO MKC-BNMO 'alternative'
!RUN DO_CP BNMO-HEMO BNMO-HEMO 'alternative'
!RUN DO_CP HEMO-STL HEMO-STL 'alternative'
..      !RUN DO_CP STL-S_SINK STL-S_SINK 'alternative'

!RUN SYSTEM 'alternative'

FI
ENDMACRO

MACRO DO_PRM1 prm_dssfn pf_dssfn mpk_dssfn ts_dssfn alternative start_date end_date
!RUN OPEN prm_dssfn pf_dssfn mpk_dssfn ts_dssfn start_date end_date
!RUN DO_RES FTPK FTPK-GARR FTPK-FTPKR FTPKR-SUX 'alternative'
!RUN DO_CP SUX-OMA SUX-S_SINK 'alternative'
!RUN SYSTEM 'alternative'
FI
ENDMACRO

MACRO DO_MRD mrd_dssfn pf_dssfn mpk_dssfn ts_dssfn alternative start_date end_date
!RUN OPEN mrd_dssfn pf_dssfn mpk_dssfn ts_dssfn start_date end_date

!RUN DO_RES FTPK FTPK-GARR FTPK-GARR FTPK-GARR 'alternative'
!RUN DO_RES GARR GARR-OAHE GARR-OAHE GARR-OAHE 'alternative'
!RUN DO_RES OAHE OAHE-BEND OAHE-BEND OAHE-BEND 'alternative'
!RUN DO_RES BEND BEND-FTRA BEND-FTRA BEND-FTRA 'alternative'
!RUN DO_RES FTRA FTRA-GAPT FTRA-GAPT FTRA-GAPT 'alternative'

!RUN DO_CP SUX-OMA SUX-OMA 'alternative'
!RUN DO_CP OMA-NCNE OMA-NCNE 'alternative'
!RUN DO_CP NCNE-MKC NCNE-MKC 'alternative'
!RUN DO_CP MKC-BNMO MKC-BNMO 'alternative'
!RUN DO_CP BNMO-HEMO BNMO-HEMO 'alternative'
!RUN DO_CP HEMO-STL HEMO-STL 'alternative'
..      !RUN DO_CP STL-S_SINK STL-S_SINK 'alternative'
ENDMACRO
```

MACRO OPEN1 prm_dssfn start_yr end_yr

```
!RUN OPEN prm_dssfn PENCMP prm_dssfnO MRD 31MARstart_yr 28FEBend_yr
ENDMACRO
```

MACRO OPEN prm_dssfn pf_dssfn mpk_dssfn ts_dssfn start_date end_date

```
OP prm_dssfn    1
OP pf_dssfn     2
OP mpk_dssfn    3
OP ts_dssfn     4
OP HECPRM       5

TI start_date 2400 end_date 2400

..        Do not extrapolate when interpolating
SET.O FU=TABLE IEXTR=1

SET.F PL=46
CLEAR
TA.F-A V=
TA.FA
GET IXMON ///MON//1MON//  5
GET FACTOR /FACTOR//CFS-KAF//1MON//  5
DEF.T E=1MON V=PENTMP
SET.V V=IXMON FO=(I6)
ENDMACRO
```

MACRO GET_TS var_name river node parameter alternative file_index

```
GET var_name A=river B=node C=parameter E=1MON F=alternative file_index
COM var_name_CFS=var_name/FACTOR
SET.V V=var_name_CFS U=CFS C=parameter(CFS)
ENDMACRO
```

MACRO MONVARY partB parameter category indep_var depend_var pfalter file_index

```
..       12 months of data in one pathname

PURGE V=depend_var
DEF.T E=1MON V=depend_var
COM depend_var=0

PURGE V=PNLTY_FUNC
GET PNLTY_FUNC A= B=partB C=parameter-PNLTY_category D= E= F=pfalter file_index

COM IF(IXMON EQ  1) depend_var=TABLE(indep_var, PNLTY_FUNC(,1), PNLTY_FUNC(, 2))
COM IF(IXMON EQ  2) depend_var=TABLE(indep_var, PNLTY_FUNC(,1), PNLTY_FUNC(, 3))
COM IF(IXMON EQ  3) depend_var=TABLE(indep_var, PNLTY_FUNC(,1), PNLTY_FUNC(, 4))
COM IF(IXMON EQ  4) depend_var=TABLE(indep_var, PNLTY_FUNC(,1), PNLTY_FUNC(, 5))
COM IF(IXMON EQ  5) depend_var=TABLE(indep_var, PNLTY_FUNC(,1), PNLTY_FUNC(, 6))
COM IF(IXMON EQ  6) depend_var=TABLE(indep_var, PNLTY_FUNC(,1), PNLTY_FUNC(, 7))
COM IF(IXMON EQ  7) depend_var=TABLE(indep_var, PNLTY_FUNC(,1), PNLTY_FUNC(, 8))
COM IF(IXMON EQ  8) depend_var=TABLE(indep_var, PNLTY_FUNC(,1), PNLTY_FUNC(, 9))
COM IF(IXMON EQ  9) depend_var=TABLE(indep_var, PNLTY_FUNC(,1), PNLTY_FUNC(,10))
COM IF(IXMON EQ 10) depend_var=TABLE(indep_var, PNLTY_FUNC(,1), PNLTY_FUNC(,11))
COM IF(IXMON EQ 11) depend_var=TABLE(indep_var, PNLTY_FUNC(,1), PNLTY_FUNC(,12))
COM IF(IXMON EQ 12) depend_var=TABLE(indep_var, PNLTY_FUNC(,1), PNLTY_FUNC(,13))
ENDMACRO
```

MACRO MONVARY1 partB parameter category indep_var depend_var pfalter file_index

```
..Penalty function is stored in a separate path for each month
..- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

PURGE V=depend_var
DEF.T E=1MON V=depend_var
COM depend_var=0

PURGE V=PJAN,PFEB,PMAR,PAPR,PMAY,PJUN,PJUL,PAUG,PSEP,POCT,PNOV,PDEC
GET PJAN A= B=partB C=parameter-PNLTY_category D= E=JAN F=pfalter file_index
GET PFEB A= B=partB C=parameter-PNLTY_category D= E=FEB F=pfalter file_index
GET PMAR A= B=partB C=parameter-PNLTY_category D= E=MAR F=pfalter file_index
GET PAPR A= B=partB C=parameter-PNLTY_category D= E=APR F=pfalter file_index
GET PMAY A= B=partB C=parameter-PNLTY_category D= E=MAY F=pfalter file_index
GET PJUN A= B=partB C=parameter-PNLTY_category D= E=JUN F=pfalter file_index
GET PJUL A= B=partB C=parameter-PNLTY_category D= E=JUL F=pfalter file_index
GET PAUG A= B=partB C=parameter-PNLTY_category D= E=AUG F=pfalter file_index
GET PSEP A= B=partB C=parameter-PNLTY_category D= E=SEP F=pfalter file_index
GET POCT A= B=partB C=parameter-PNLTY_category D= E=OCT F=pfalter file_index
GET PNOV A= B=partB C=parameter-PNLTY_category D= E=NOV F=pfalter file_index
GET PDEC A= B=partB C=parameter-PNLTY_category D= E=DEC F=pfalter file_index

COM IF(IXMON EQ  1) depend_var=TABLE(indep_var, PJAN(,1), PJAN(,2))
COM IF(IXMON EQ  2) depend_var=TABLE(indep_var, PFEB(,1), PFEB(,2))
COM IF(IXMON EQ  3) depend_var=TABLE(indep_var, PMAR(,1), PMAR(,2))
COM IF(IXMON EQ  4) depend_var=TABLE(indep_var, PAPR(,1), PAPR(,2))
COM IF(IXMON EQ  5) depend_var=TABLE(indep_var, PMAY(,1), PMAY(,2))
COM IF(IXMON EQ  6) depend_var=TABLE(indep_var, PJUN(,1), PJUN(,2))
COM IF(IXMON EQ  7) depend_var=TABLE(indep_var, PJUL(,1), PJUL(,2))
COM IF(IXMON EQ  8) depend_var=TABLE(indep_var, PAUG(,1), PAUG(,2))
COM IF(IXMON EQ  9) depend_var=TABLE(indep_var, PSEP(,1), PSEP(,2))
COM IF(IXMON EQ 10) depend_var=TABLE(indep_var, POCT(,1), POCT(,2))
COM IF(IXMON EQ 11) depend_var=TABLE(indep_var, PNOV(,1), PNOV(,2))
COM IF(IXMON EQ 12) depend_var=TABLE(indep_var, PDEC(,1), PDEC(,2))
ENDMACRO
```

MACRO MONSAME partB parameter category indep_var depend_var pfalter file_index

```
PURGE V=depend_var
DEF.T E=1MON V=depend_var
COM depend_var=0

PURGE V=PNLTY_FUNC
GET PNLTY_FUNC A= B=partB C=parameter-PNLTY_category D= E= F=pfalter file_index
COM depend_var=TABLE(indep_var,PNLTY_FUNC(,1),PNLTY_FUNC(,2))
ENDMACRO
```

MACRO CLASSDEF start end interval

```
COM NU_CLASS = end - start / interval + 1
PURGE V=CLASS
DEF.P V=CLASS(NU_CLASS,2)
COM CLASS(,1)=1
COM CLASS(,2)=interval
COM CLASS(1,2)=start
COM CLASS=ACCUM(CLASS)
ENDMACRO
```

MACRO COUNTLO parameter min_value count_mo count_yr

```
PURGE V=T_PARAM, count_yr
DEF.T E=1MON V=count_mo, T_PARAM
COM count_mo=0
COM T_PARAM=parameter
COM IF(SEASON LT .5) T_PARAM=1.0E+20
COM IF(T_PARAM LT min_value) count_mo=1
COM count_yr=DERIVE(count_mo,1YEAR,SUM)
COM IF(SEASON LT .5) T_PARAM=parameter
ENDMACRO
```

MACRO RES_STAT

```
DEF.P V=RES_STAT(5,16)

!RUN STAT1 STORAGE        1    RES_STAT
!RUN STAT1 ELEV_POOL      2    RES_STAT
!RUN STAT1 RELEASE        3    RES_STAT
!RUN STAT1 RELEASE_CFS    4    RES_STAT
!RUN STAT1 HPC_POOL       5    RES_STAT
!RUN STAT1 REC_POOL       6    RES_STAT
!RUN STAT1 WSP_POOL       7    RES_STAT
!RUN STAT1 CMP_POOL       8    RES_STAT
!RUN STAT1 EDT_POOL       9    RES_STAT
!RUN STAT1 DIFF_POOL     10    RES_STAT
!RUN STAT1 HPE_REL       11    RES_STAT
!RUN STAT1 HPE_EDT_REL   12    RES_STAT
!RUN STAT1 REC_REL       13    RES_STAT
!RUN STAT1 CMP_REL       14    RES_STAT
!RUN STAT1 EDT_REL       15    RES_STAT
!RUN STAT1 DIFF_REL      16    RES_STAT

SET.V V=RES_STAT U1= ,U2= ,T1= ,T2= ,C=

SET.VL V=RES_STAT 1=ELEV_POOL
SET.VL V=RES_STAT 2=RELEASE
SET.VL V=RES_STAT 3=RELEASE_CFS
SET.VL V=RES_STAT 4=HPC_POOL
SET.VL V=RES_STAT 5=REC_POOL
SET.VL V=RES_STAT 6=WSP_POOL
SET.VL V=RES_STAT 7=CMP_POOL
SET.VL V=RES_STAT 8=EDT_POOL
SET.VL V=RES_STAT 9=DIFF_POOL
SET.VL V=RES_STAT 10=HPE_REL
SET.VL V=RES_STAT 11=HPE_EDT_REL
SET.VL V=RES_STAT 12=REC_REL
SET.VL V=RES_STAT 13=CMP_REL
SET.VL V=RES_STAT 14=EDT_REL
SET.VL V=RES_STAT 15=DIFF_REL
SET.F P=(I14,2X,2F10.1,F10.2,I8,14F10.3)
ENDMACRO
```

MACRO CP_STAT

```
DEF.P V=CP_STAT(5,10)

!RUN STAT1 FLOW          1    CP_STAT
!RUN STAT1 FLOW_CFS      2    CP_STAT
!RUN STAT1 FDA_FLOW      3    CP_STAT
!RUN STAT1 FDU_FLOW      4    CP_STAT
!RUN STAT1 NAV_FLOW      5    CP_STAT
!RUN STAT1 REC_FLOW      6    CP_STAT
!RUN STAT1 WSP_FLOW      7    CP_STAT
!RUN STAT1 CMP_FLOW      8    CP_STAT
!RUN STAT1 EDT_FLOW      9    CP_STAT
!RUN STAT1 DIFF_FLOW    10    CP_STAT

SET.V V=CP_STAT U1= ,U2= ,T1= ,T2= ,C=

SET.VL V=CP_STAT 1=FLOW_CFS
SET.VL V=CP_STAT 2=FDA_FLOW
SET.VL V=CP_STAT 3=FDU_FLOW
SET.VL V=CP_STAT 4=NAV_FLOW
SET.VL V=CP_STAT 5=REC_FLOW
SET.VL V=CP_STAT 6=WSP_FLOW
SET.VL V=CP_STAT 7=CMP_FLOW
SET.VL V=CP_STAT 8=EDT_FLOW
SET.VL V=CP_STAT 9=DIFF_FLOW

SET.F P=(I14,2X,F10.2,I8,8F10.3)
ENDMACRO
```

MACRO STAT1 variable index var_out

```
COM var_out(1,index)=MIN(variable)
COM var_out(2,index)=MAX(variable)
COM var_out(3,index)=MEAN(variable)
COM var_out(4,index)=SD(variable)
COM var_out(5,index)=SUM(variable)
ENDMACRO
```

```
MACRO DO_RES res_node rel_loc hpe_loc relo_loc alternative
```

```
!RUN DO_RES1 'res_node' 'rel_loc' 'hpe_loc' 'relo_loc' 'alternative'
!RUN RES_STAT
!RUN DO_RES2 'res_node'
!RUN DO_RES3 'res_node' 'rel_loc' 'alternative'
!RUN RES_DUR 'res_node' 'rel_loc' 'hpe_loc' 'alternative'
ENDMACRO
```

```
MACRO DO_RES1 res_node rel_loc hpe_loc relo_loc alternative
```

```
..                                          Get release & storage

GET STORAGE A= B=res_node C=STOR E=1MON F=alternative 1
GET RELEASE A= B=relo_loc C=FLOW D= E=1MON F=alternative 1
COM RELEASE_CFS=RELEASE/FACTOR
SET.V V=RELEASE_CFS U=CFS C=FLOW(CFS)

PURGE V=ELARCA
GET ELARCA A= B=res_node C=EL-AR-CAP D= E= F= 2
DEF.T E=1MON V=ELEV_POOL
COM ELEV_POOL=TABLE(STORAGE,ELARCA(,3),ELARCA(,1))
SET.V V=ELEV_POOL U=FT T=INST-VAL
..
!RUN MONVARY  'res_node' STOR HPC STORAGE HPC_POOL '       ' 2
!RUN MONVARY  'res_node' STOR REC STORAGE REC_POOL '       ' 2
!RUN MONVARY  'res_node' STOR WSP STORAGE WSP_POOL '       ' 2
!RUN MONVARY  'res_node' STOR CMP STORAGE CMP_POOL '       ' 2
!RUN MONVARY1 'res_node' STOR EDT STORAGE EDT_POOL 'alternative' 1

!RUN MONSAME  'rel_loc' FLOW HPE RELEASE HPE_REL        '       '    2
!RUN MONVARY1 'hpe_loc' FLOW HPE_EDT RELEASE HPE_EDT_REL  'alternative' 1
!RUN MONVARY  'rel_loc' FLOW REC RELEASE REC_REL        '       '    2
!RUN MONVARY  'rel_loc' FLOW CMP RELEASE CMP_REL        '       '    2
!RUN MONVARY1 'relo_loc' FLOW EDT RELEASE EDT_REL       'alternative' 1

COM DIFF_POOL=EDT_POOL-CMP_POOL
COM DIFF_REL=EDT_REL-CMP_REL
ENDMACRO
```

MACRO DO_RES2 res_node

```
SET.F T=(F10.1)
SET.V V=STORAGE      FO=(F10.1)
SET.V V=ELEV_POOL    FO=(F10.1)  U=FT
SET.V V=RELEASE      FO=(F10.2)
SET.V V=RELEASE_CFS FO=(I8)      U=CFS

SELECT
STORAGE
ELEV_POOL
RELEASE
RELEASE_CFS
HPC_POOL
REC_POOL
WSP_POOL
CMP_POOL
EDT_POOL
DIFF_POOL
HPE_REL
HPE_EDT_REL
REC_REL
CMP_REL
EDT_REL
DIFF_REL
END
SET.F TILN=1 TI=res_node
SET.F TILN=2 TI=(Reservoir storage, elevation, release, and penalties)

ST.O
TAB.A

SET.F TILN=0
SET.F PFLN=1 PFL='Minimum       '
SET.F PFLN=2 PFL='Maximum       '
SET.F PFLN=3 PFL='Mean          '
SET.F PFLN=4 PFL='Std. Deviation'
SET.F PFLN=5 PFL='Sum           '
TAB.A V=RES_STAT
SET.F PFLN=0
ENDMACRO
```

```
MACRO DO_RES3 res_node rel_loc alternative

SET.V V=STORAGE      U=KAF T=INST-VAL A=  B=res_node C=STOR              F=alternative
SET.V V=ELEV_POOL    U=FT  T=INST-VAL A=  B=res_node C=ELEV              F=alternative
SET.V V=RELEASE      U=KAF T=PER-AVER A=  B=rel_loc  C=FLOW              F=alternative
SET.V V=RELEASE_CFS  U=CFS T=PER-AVER A=  B=rel_loc C=FLOW(CFS)          F=alternative
SET.V V=HPC_POOL     U=K$  T=PER-AVER A=  B=res_node C=PNLTY_HPC         F=alternative
SET.V V=REC_POOL     U=K$  T=PER-AVER A=  B=res_node C=PNLTY_REC         F=alternative
SET.V V=WSP_POOL     U=K$  T=PER-AVER A=  B=res_node C=PNLTY_WSP         F=alternative
SET.V V=CMP_POOL     U=K$  T=PER-AVER A=  B=res_node C=PNLTY_CMP         F=alternative
SET.V V=EDT_POOL     U=K$  T=PER-AVER A=  B=res_node C=PNLTY_EDT         F=alternative
SET.V V=DIFF_POOL    U=K$  T=PER-AVER A=  B=res_node C=PNLTY_DIFF        F=alternative
SET.V V=HPE_REL      U=K$  T=PER-AVER A=  B=rel_loc C=PNLTY_HPE     F=alternative
SET.V V=HPE_EDT_REL  U=K$  T=PER-AVER A=  B=rel_loc C=PNLTY_HPE_EDT F=alternative
SET.V V=REC_REL      U=K$  T=PER-AVER A=  B=rel_loc C=PNLTY_REC     F=alternative
SET.V V=CMP_REL      U=K$  T=PER-AVER A=  B=rel_loc C=PNLTY_CMP     F=alternative
SET.V V=EDT_REL      U=K$  T=PER-AVER A=  B=rel_loc C=PNLTY_EDT     F=alternative
SET.V V=DIFF_REL     U=K$  T=PER-AVER A=  B=rel_loc C=PNLTY_DIFF    F=alternative

PUT STORAGE      3
PUT ELEV_POOL    3
PUT RELEASE      3
PUT RELEASE_CFS  3
PUT HPC_POOL     3
PUT REC_POOL     3
PUT WSP_POOL     3
PUT CMP_POOL     3
PUT EDT_POOL     3
PUT DIFF_POOL    3
PUT HPE_REL      3
PUT HPE_EDT_REL  3
PUT REC_REL      3
PUT CMP_REL      3
PUT EDT_REL      3
PUT DIFF_REL     3
ENDMACRO
```

MACRO DO_CP loc_norm loc_job alternative

```
!RUN DO_CP1 'loc_norm' 'loc_job' 'alternative'
!RUN CP_STAT
!RUN DO_CP2 'loc_norm'
!RUN DO_CP3 'loc_norm' 'alternative'
!RUN CP_DUR 'loc_norm' 'loc_job' 'alternative'
ENDMACRO
```

MACRO DO_CP1 loc_norm loc_job alternative

```
GET FLOW A= B=loc_job C=FLOW D= E=1MON F=alternative 1
COM FLOW_CFS=FLOW/FACTOR
SET.V V=FLOW_CFS U=CFS C=FLOW(CFS)

!RUN MONVARY 'loc_norm' FLOW FDA FLOW FDA_FLOW '        ' 2
!RUN MONVARY 'loc_norm' FLOW FDU FLOW FDU_FLOW '        ' 2
!RUN MONVARY 'loc_norm' FLOW NAV FLOW NAV_FLOW '        ' 2
!RUN MONVARY 'loc_norm' FLOW REC FLOW REC_FLOW '        ' 2
!RUN MONVARY 'loc_norm' FLOW WSP FLOW WSP_FLOW '        ' 2
!RUN MONVARY 'loc_norm' FLOW CMP FLOW CMP_FLOW '        ' 2
!RUN MONVARY1 'loc_job' FLOW EDT FLOW EDT_FLOW 'alternative' 1

COM DIFF_FLOW=EDT_FLOW-CMP_FLOW
ENDMACRO
```

MACRO DO_CP2 loc_norm

```
PENTMP=0
SET.F T=(F10.1)
SET.V V=FLOW        FO=(F10.2)
SET.V V=FLOW_CFS    FO=(I8)

SELECT
FLOW
FLOW_CFS
FDA_FLOW
FDU_FLOW
NAV_FLOW
REC_FLOW
WSP_FLOW
CMP_FLOW
EDT_FLOW
DIFF_FLOW
END

SET.F TILN=1 TI=loc_norm
SET.F TILN=2 TI=(Flow and penalty)
TAB.A

SET.F PFLN=1 PFL='Minimum       '
SET.F PFLN=2 PFL='Maximum       '
SET.F PFLN=3 PFL='Mean          '
SET.F PFLN=4 PFL='Std. Deviation'
SET.F PFLN=5 PFL='Sum           '

SET.F TILN=0
TAB.A V=CP_STAT
SET.F PFLN=0
ENDMACRO
```

MACRO DO_CP3 loc_norm alternative

```
SET.V V=FLOW        U=KAF T=PER-AVER A= B=loc_norm C=FLOW        F=alternative
SET.V V=FLOW_CFS    U=KAF T=PER-AVER A= B=loc_norm C=FLOW        F=alternative
SET.V V=FDA_FLOW    U=CFS T=PER-AVER A= B=loc_norm C=FLOW(CFS)   F=alternative
SET.V V=FDU_FLOW    U=K$  T=PER-AVER A= B=loc_norm C=PNLTY_FDU   F=alternative
SET.V V=NAV_FLOW    U=K$  T=PER-AVER A= B=loc_norm C=PNLTY_NAV   F=alternative
SET.V V=REC_FLOW    U=K$  T=PER-AVER A= B=loc_norm C=PNLTY_REC   F=alternative
SET.V V=WSP_FLOW    U=K$  T=PER-AVER A= B=loc_norm C=PNLTY_WSP   F=alternative
SET.V V=CMP_FLOW    U=K$  T=PER-AVER A= B=loc_norm C=PNLTY_CMP   F=alternative
SET.V V=EDT_FLOW    U=K$  T=PER-AVER A= B=loc_norm C=PNLTY_EDT   F=alternative
SET.V V=DIFF_FLOW   U=K$  T=PER-AVER A= B=loc_norm C=PNLTY_DIFF  F=alternative

PUT FLOW      3
PUT FLOW_CFS  3
PUT FDA_FLOW  3
PUT FDU_FLOW  3
PUT NAV_FLOW  3
PUT REC_FLOW  3
PUT WSP_FLOW  3
PUT CMP_FLOW  3
PUT EDT_FLOW  3
PUT DIFF_FLOW 3
ENDMACRO
```

MACRO SYSTEM alternative

```
CLEAR
..
GET IN_FTPK   //FTPK/FLOW_LOC_INC//1MON// 4
GET DPL_FTPK  //FTPK/FLOW_LOC_DPL//1MON// 4
GET EVAP_FTPK //FTPK/EVAP//1MON/alternative/ 1
GET STOR_FTPK //FTPK/STOR//1MON/alternative/ 1
..
GET IN_GARR   //GARR/FLOW_LOC_INC//1MON// 4
GET DPL_GARR  //GARR/FLOW_LOC_DPL//1MON// 4
GET EVAP_GARR //GARR/EVAP//1MON/alternative/ 1
GET STOR_GARR //GARR/STOR//1MON/alternative/ 1
..
GET IN_OAHE   //OAHE/FLOW_LOC_INC//1MON// 4
GET DPL_OAHE  //OAHE/FLOW_LOC_DPL//1MON// 4
GET EVAP_OAHE //OAHE/EVAP//1MON/alternative/ 1
GET STOR_OAHE //OAHE/STOR//1MON/alternative/ 1
..
GET EVAP_BEND //BEND/EVAP//1MON/alternative/ 1
GET STOR_BEND //BEND/STOR//1MON/alternative/ 1
..
GET IN_FTRA   //FTRA/FLOW_LOC_INC//1MON// 4
GET DPL_FTRA  //FTRA/FLOW_LOC_DPL//1MON// 4
GET EVAP_FTRA //FTRA/EVAP//1MON/alternative/ 1
GET STOR_FTRA //FTRA/STOR//1MON/alternative/ 1
..
GET IN_GAPT   //GAPT/FLOW_LOC_INC//1MON// 4
GET DPL_GAPT  //GAPT/FLOW_LOC_DPL//1MON// 4
GET EVAP_GAPT //GAPT/EVAP//1MON/alternative/ 1
GET STOR_GAPT //GAPT/STOR//1MON/alternative/ 1
..
DEF.T E=1MON V=IN_SYS,DPL_SYS,EVAP_SYS,STOR_SYS,
DEF.T E=1MON V=APOW_SYS,PPOW_SYS,ENRGY_SYS,NAV_LVL,NAV_LNGTH
..
COM IF(IN_FTPK    LE -1E+29) IN_FTPK    =0.0
COM IF(DPL_FTPK   LE -1E+29) DPL_FTPK   =0.0
COM IF(EVAP_FTPK  LE -1E+29) EVAP_FTPK  =0.0
COM IF(STOR_FTPK  LE -1E+29) STOR_FTPK  =0.0
..
COM IF(IN_GARR    LE -1E+29) IN_GARR    =0.0
COM IF(DPL_GARR   LE -1E+29) DPL_GARR   =0.0
COM IF(EVAP_GARR  LE -1E+29) EVAP_GARR  =0.0
COM IF(STOR_GARR  LE -1E+29) STOR_GARR  =0.0
..
COM IF(IN_OAHE    LE -1E+29) IN_OAHE    =0.0
COM IF(DPL_OAHE   LE -1E+29) DPL_OAHE   =0.0
COM IF(EVAP_OAHE  LE -1E+29) EVAP_OAHE  =0.0
COM IF(STOR_OAHE  LE -1E+29) STOR_OAHE  =0.0
..
COM IF(EVAP_BEND  LE -1E+29) EVAP_BEND  =0.0
COM IF(STOR_BEND  LE -1E+29) STOR_BEND  =0.0
..
COM IF(IN_FTRA    LE -1E+29) IN_FTRA    =0.0
COM IF(DPL_FTRA   LE -1E+29) DPL_FTRA   =0.0
COM IF(EVAP_FTRA  LE -1E+29) EVAP_FTRA  =0.0
COM IF(STOR_FTRA  LE -1E+29) STOR_FTRA  =0.0
..
COM IF(IN_GAPT    LE -1E+29) IN_GAPT    =0.0
COM IF(DPL_GAPT   LE -1E+29) DPL_GAPT   =0.0
COM IF(EVAP_GAPT  LE -1E+29) EVAP_GAPT  =0.0
COM IF(STOR_GAPT  LE -1E+29) STOR_GAPT  =0.0

COM IN_SYS=IN_FTPK+IN_GARR+IN_OAHE+IN_FTRA+IN_GAPT
COM DPL_SYS=DPL_FTPK+DPL_GARR+DPL_OAHE+DPL_FTRA+DPL_GAPT
COM EVAP_SYS=EVAP_FTPK+EVAP_GARR+EVAP_OAHE+EVAP_BEND+EVAP_FTRA+EVAP_GAPT
COM STOR_SYS=STOR_FTPK+STOR_GARR+STOR_OAHE+STOR_BEND+STOR_FTRA+STOR_GAPT
..
TAB.F V=EVAP_FTPK, STOR_FTPK, EVAP_GARR, STOR_GARR
TAB.F V=IN_SYS,EVAP_SYS,DPL_SYS,STOR_SYS
TAB.F V=STOR_FTPK,STOR_GARR,STOR_OAHE,STOR_BEND,STOR_FTRA,STOR_GAPT,STOR_SYS
..
ST.V4
CLEAR
ENDMACRO
```

MACRO RES_DUR res_node rel_loc relo_loc alternative

```
!RUN DURATION 'res_node' 'res_node' ELEV 'alternative'
PUT DUR        3
PUT LACCUMDUR  3
PUT HACCUMDUR  3
PUT PCTEXCEED  3

!RUN DURATION 'rel_loc' 'rel_loc' FLOW 'alternative'
PUT DUR       3
PUT LACCUMDUR 3
PUT HACCUMDUR 3
PUT PCTEXCEED 3

ENDMACRO
```

MACRO CP_DUR loc_norm loc_job alternative

```
!RUN DURATION 'loc_norm' 'loc_job' FLOW 'alternative'
PUT DUR        3
PUT LACCUMDUR  3
PUT HACCUMDUR  3
PUT PCTEXCEED  3

ENDMACRO
```

```
MACRO DURATION loc_norm loc_job parameter alternative
```

```
PURGE V=CLASS,PARAMETER,DUR,LACCUMDUR,TOTLDUR,HACCUMDUR,PCTEXCEED,DUR_SUMRY
..
GET CLASS //loc_norm/INDEX-parameter_CLASS_INTRVL////  2
GET PARAMETER //loc_norm/parameter//IMON/alternative/  3

DEF.C V=NU_CLASS,KU_CLASS,MAX_PARM,MAX_CLASS,MAX_CLASS1

COM NU_CLASS=COUNT(CLASS(,1))
COM KU_CLASS=NU_CLASS
COM MAX_PARM=MAX(PARAMETER)
COM MAX_CLASS=MAX(CLASS(,2))
COM IF(MAX_PARM.GT.MAX_CLASS) NU_CLASS=NU_CLASS+1
COM MAX_CLASS1=ABS(MAX_CLASS)
COM MAX_CLASS1=.01*MAX_CLASS1+MAX_CLASS

DEF.P V=DUR(NU_CLASS,14)
SET.VL V=DUR L=JAN FEB MAR APR MAY JUN JUL AUG SEP OCT NOV DEC TOT

COM DUR(,1)=CLASS(,2)
COM IF(NU_CLASS.GT.KU_CLASS) DUR(NU_CLASS,1)=MAX_CLASS
DEF.P V=DUR_SUMRY(NU_CLASS,5)

COM IF(IXMON.EQ. 1) DUR(, 2)=DURATION(PARAMETER,DUR(,1))
COM IF(IXMON.EQ. 2) DUR(, 3)=DURATION(PARAMETER,DUR(,1))
COM IF(IXMON.EQ. 3) DUR(, 4)=DURATION(PARAMETER,DUR(,1))
COM IF(IXMON.EQ. 4) DUR(, 5)=DURATION(PARAMETER,DUR(,1))
COM IF(IXMON.EQ. 5) DUR(, 6)=DURATION(PARAMETER,DUR(,1))
COM IF(IXMON.EQ. 6) DUR(, 7)=DURATION(PARAMETER,DUR(,1))
COM IF(IXMON.EQ. 7) DUR(, 8)=DURATION(PARAMETER,DUR(,1))
COM IF(IXMON.EQ. 8) DUR(, 9)=DURATION(PARAMETER,DUR(,1))
COM IF(IXMON.EQ. 9) DUR(,10)=DURATION(PARAMETER,DUR(,1))
COM IF(IXMON.EQ.10) DUR(,11)=DURATION(PARAMETER,DUR(,1))
COM IF(IXMON.EQ.11) DUR(,12)=DURATION(PARAMETER,DUR(,1))
COM IF(IXMON.EQ.12) DUR(,13)=DURATION(PARAMETER,DUR(,1))
COM DUR(,14)=DURATION(PARAMETER,DUR(,1))

DEF.P V=LACCUMDUR(NU_CLASS,14), TOTLDUR(1,14)
DEF.P V=HACCUMDUR(NU_CLASS,14), PCTEXCEED(NU_CLASS,14)

COM LACCUMDUR(,1)=DUR(,1)
COM TOTLDUR(,1)=0
COM HACCUMDUR(,1)=DUR(,1)
COM PCTEXCEED(,1)=DUR(,1)

COM LACCUMDUR(,2-14)=ACCUM(DUR(,2-14))
COM TOTLDUR(,2-14)=SUM(DUR(,2-14))
COM HACCUMDUR(,2-14)=TOTLDUR(1,2-14)-LACCUMDUR(,2-14)
COM PCTEXCEED(,2-14)=HACCUMDUR(,2-14)/TOTLDUR(,2-14) * 100.0

COM DUR_SUMRY(,1)=DUR(,1)
COM DUR_SUMRY(,2)=DUR(,14)
COM DUR_SUMRY(,3)=LACCUMDUR(,14)
COM DUR_SUMRY(,4)=HACCUMDUR(,14)
COM DUR_SUMRY(,5)=PCTEXCEED(,14)

SET.VL V=DUR        L=JAN FEB MAR APR MAY JUN JUL AUG SEP OCT NOV DEC TOT
SET.V V=DUR A= B=loc_norm C=parameter-NU_INCLASS D= E= F=alternative
SET.V V=DUR IHORIZ=1

SET.VL V=TOTLDUR    L=JAN FEB MAR APR MAY JUN JUL AUG SEP OCT NOV D
SET.V  V=TOTLDUR A= B=loc_norm C=parameter-NU_INCLASS_TOT D= E= F=alternative
SET.V  V=TOTLDUR IHORIZ=1

SET.VL V=LACCUMDUR  L=JAN FEB MAR APR MAY JUN JUL AUG SEP OCT NOV DEC TOT
SET.V V=LACCUMDUR A= B=loc_norm C=parameter-NU_INCLASS_CUMLO D= E= F=alternative
SET.V V=LACCUMDUR IHORIZ=1

SET.VL V=HACCUMDUR  L=JAN FEB MAR APR MAY JUN JUL AUG SEP OCT NOV DEC TOT
SET.V V=HACCUMDUR A= B=loc_norm C=parameter-NU_INCLASS_CUMHI D= E= F=alternative
SET.V V=HACCUMDUR IHORIZ=1

SET.VL V=PCTEXCEED  L=JAN FEB MAR APR MAY JUN JUL AUG SEP OCT NOV DEC TOT
SET.V V=PCTEXCEED A= B=loc_norm C=parameter-NU_INCLASS_CUMHI_PCT D= E= F=alternative
SET.V V=PCTEXCEED IHORIZ=1

SET.VL V=DUR_SUMRY  L=HITS,'L ACCUM', 'H ACCUM','EXCEED %'
SET.V V=DUR_SUMRY   A= B=loc_norm C=parameter-NU_INCLASS_SUMRY D= E= F=alternative
SET.V V=DUR_SUMRY   IHORIZ=1

ENDMACRO
```

MACRO POWER res_node node_coeff relo_loc alternative

```
PURGE V=POW_STORAGE,RELEASE,RELEASE_CFS,ENRGYCOEFF,ENERGY,CAPACITY
GET POW_STORAGE A= B=res_node C=STOR E=1MON F=alternative 1
GET RELEASE A= B=relo_loc C=FLOW D= E=1MON F=alternative 1
COM RELEASE_CFS=RELEASE/FACTOR
COM RELEASE_KCFS=RELEASE_CFS/1000
SET.V V=RELEASE_KCFS U=KCFS C=FLOW(CFS)

PURGE V=ELARCA
GET ELARCA A= B=res_node C=EL-AR-CAP D= E= F= 2
DEF.T E=1MON V=ELEV_POOL
COM ELEV_POOL=TABLE(POW_STORAGE,ELARCA(,3),ELARCA(,1))
SET.V V=ELEV_POOL U=FT T=INST-VAL

GET ENRGYCOEFF //node_coeff/EL-COEFF_ENRGY//// 2

COM A=TABLEM(ELEV_POOL,ENRGYCOEFF(,1),ENRGYCOEFF(,2))
COM B=TABLEM(ELEV_POOL,ENRGYCOEFF(,1),ENRGYCOEFF(,3))
COM CAPACITY=TABLE(ELEV_POOL,ENRGYCOEFF(,1),ENRGYCOEFF(,4))
COM ENERGY=RELEASE_KCFS*A+B

ENDMACRO
```

MACRO POWER_P res_node node_coeff relo_loc alternative

```
SET.V V=ENERGY U=KMWH T=UNT  A= B=node_coeff C=ENERGY F=
SET.V V=CAPACITY U=MW T=UNT  A= B=node_coeff C=POWER_PEAK F=
PUT CAPACITY     3
PUT ENERGY       3
ENDMACRO
```

```
MACRO POWER1 res_node relo_loc alternative
```

```
PURGE V=RELEASE,RELEASE_CFS
GET RELEASE A= B=relo_loc C=FLOW D= E=1MON F=alternative 1
COM RELEASE_CFS=RELEASE/FACTOR
COM RELEASE_KCFS=RELEASE_CFS/1000
SET.V V=RELEASE_KCFS U=KCFS C=FLOW(CFS)

GET ENRGYCOEFF //res_node/FLOW-POWER//// 2

COM ENERGY=TABLE(RELEASE_KCFS,ENRGYCOEFF(,1),ENRGYCOEFF(,2))
COM CAPACITY=TABLE(RELEASE_KCFS,ENRGYCOEFF(,3),ENRGYCOEFF(,4))

ENDMACRO
```

```
MACRO POWER1_P res_node relo_loc alternative
```

```
SET.V V=ENERGY U=KMWH T=UNT  A= B=node_coeff C=ENERGY F=
SET.V V=CAPACITY U=MW T=UNT  A= B=node_coeff C=POWER_PEAK F=
PUT CAPACITY     3
PUT ENERGY       3
ENDMACRO
```

```
MACRO FLOW_LOC id_depletion
```

```
!RUN Q_LOC1 FTPK id_depletion
!RUN Q_LOC1 GARR id_depletion
!RUN Q_LOC1 OAHE id_depletion
..     !RUN Q_LOC1 BEND id_depletion
!RUN Q_LOC1 FTRA id_depletion
!RUN Q_LOC1 GAPT id_depletion
!RUN Q_LOC1 SUX  id_depletion
!RUN Q_LOC1 OMA  id_depletion
!RUN Q_LOC1 NCNE id_depletion
!RUN Q_LOC1 MKC  id_depletion
!RUN Q_LOC1 BNMO id_depletion
!RUN Q_LOC1 HEMO id_depletion
ENDMACRO
```

```
MACRO Q_LOC1 location id_depletion
```

```
PURGE V=Q_ADJ,Q_DPL,Q_LOC
GET Q_LOC //location/FLOW_LOC_INC//1MON// 1
GET Q_DPL A=  B=location  C=FLOW_LOC_DPL  E=1MON  F=id_depletion 1
GET QM_DPL A= B=location C=MON-FLOW_DPL D=  E=  F=  1
GET MONTH A=  B=  C=MON  E=1MON F= 2
COMP Q_ADJ=Q_LOC+Q_DPL
..
COM IF(MONTH EQ  1) Q_ADJ=Q_ADJ-QM_DPL( 1,2)
COM IF(MONTH EQ  2) Q_ADJ=Q_ADJ-QM_DPL( 2,2)
COM IF(MONTH EQ  3) Q_ADJ=Q_ADJ-QM_DPL( 3,2)
COM IF(MONTH EQ  4) Q_ADJ=Q_ADJ-QM_DPL( 4,2)
COM IF(MONTH EQ  5) Q_ADJ=Q_ADJ-QM_DPL( 5,2)
COM IF(MONTH EQ  6) Q_ADJ=Q_ADJ-QM_DPL( 6,2)
COM IF(MONTH EQ  7) Q_ADJ=Q_ADJ-QM_DPL( 7,2)
COM IF(MONTH EQ  8) Q_ADJ=Q_ADJ-QM_DPL( 8,2)
COM IF(MONTH EQ  9) Q_ADJ=Q_ADJ-QM_DPL( 9,2)
COM IF(MONTH EQ 10) Q_ADJ=Q_ADJ-QM_DPL(10,2)
COM IF(MONTH EQ 11) Q_ADJ=Q_ADJ-QM_DPL(11,2)
COM IF(MONTH EQ 12) Q_ADJ=Q_ADJ-QM_DPL(12,2)
..
SET.V V=Q_ADJ U=KAF C=FLOW_LOC F=
PUT Q_ADJ 1
ENDMACRO
```

Appendix D

Proposed Hydropower Algorithm for HEC-PRM

Proposed Hydropower Algorithm for HEC-PRM

The steps of the hydropower algorithm for HEC-PRM are as follows.

STEP A (Initialize):

- Set ITER (iteration counter) = 0.

- Set ITMAX = the maximum number of iterations allowed (must be > 1).

- Set CANDPEN (candidate optimal objective function value) = a very large number.

- Set $\Delta R_{max} = 0.50$.

  [The Mar./Apr. 1991 issue of the ASCE *Journal of Water Resources Planning and Management* (vol. 117, no. 2) includes the paper "Reservoir operations by successive linear programming," by Tao Tao and W.C. Lennox on pgs. 274-280. In this paper, these researchers conclude "... search speed can be improved by simply taking the initial step sizes to be one half of the differences of the lower and upper bounds of the variable concerned and halving the step sizes for each iteration."]

- Set $R_{j,upper}$ = release corresponding to maximum power generation at maximum head for reservoir j.

  [ΔR_{max} and $R_{j,upper}$ are used in constraining release in Step C, and are subject to change as HEC collects information on performance with alternative values].

- For each reservoir j, for each period t, estimate $S_{j,t}$, the end-of-period storage. Go to Step B.

STEP B (Set up the network):

- Set ITER = ITER +1.
- If ITER > ITMAX

  Declare the candidate solution the optimal solution and stop.

- Else if ITER ≤ ITMAX
  - Use the elevation-capacity function for reservoir j to determine the end-of-period head.
  - Average the beginning-of-period and end-of-period heads.
  - Select the "closest" user-provided linear approximation of the hydropower penalty function for each reservoir each period.
  - Set up the system network with arc bounds and costs to represent these hydropower penalty functions, along with flow and storage penalty functions for other purposes.
  - Go to Step C.

STEP C (Limit variation):

- If ITER = 1, go to Step D.
- Else if ITER > 1
  - Constrain flow on the reservoir hydropower-release links so the total release does not vary from the candidate solution by more than ΔR_{max}. The link lower bound would be $R_{j,t} * (1 - \Delta R_{max})$, and the upper bound would be $R_{j,t}(1 + \Delta R_{max})$.
  - If the candidate release is zero, set the upper bound equal $R_{j,upper}$.
  - Go to Step D.

STEP D (Solve the network):

- Solve the resulting flow-allocation problem to find CURRPEN, the penalty associated with the current approximation. Use the best available network solver at this step. If a previous network solution is available, and if the solver can use it as a starting point, let it.

- Go to Step E.

STEP E (Check for solution to nonlinear problem):

- For each reservoir j, for each period t, determine $S_{j,t-1}$ and $S_{j,t}$ from the current solution of the network.

- Do these values differ from the values used in Step B to select the approximation?

  - If all are close enough, declare the current solution optimal and stop.

  - Else, go to Step F.

STEP F (Update candidate solution):

- If CURRPEN < CANDPEN,

  it is an improvement, so save the current solution (storages, releases, etc.) as the candidate optimal solution, set CANDPEN = CURRPEN, and go to Step B.

- Else if CURRPEN ≥ CANDPEN

  go to Step G.

STEP G (Decrease the allowable variation):

- Set $\Delta R_{max} = \Delta R_{max}/2$.

- If ΔR_{max} < minimum value

  declare the candidate solution optimal and stop.

- Else

  go to Step B.

Appendix E

Workshop Agenda

Workshop Agenda

HEC-PRM Workshop For MRD
December 2-4, 1991

Day 1
Mon, December 2

8:30 - 9:00	Overview of HEC's Proposal to MRD for Phases I and II. Discussion of the current status of the MRD Study. (Burnham)
9:00 - 10:00	Overview PRM Concepts, it's role in MRD's Study. (Ford)
10:00 - 10:20	Break
10:20 - 11:45	Overview of MENUPRM and the Programs accessible from MENUPRM. Discussion of the purpose of each program and their interrelationships.
	Hands On use of RDATA0 and RDMATF (flow data input). Store results from MRD's simulation model using V1.EXE and RDMATF. Storing MRD's input data set D0DATA using RDATA0. (Carl)
1:00 - 2:00	Overview of DSS including data conventions (regular time series and paired data), programs DSSUTL, DSPLAY, DSSPD, MATHPK, and HEC-PRM. Coverage of basic files and related commands (CAtalogue.new,condensed,sorted; DisPlay pathnames, selective catalog, DUplicating pathnames, EDiting data). (Hayes)
	Description of pathname parts used in MRD study and recommended procedures for entering, transferring, and naming data.
2:15 - 4:30	Hands-On Application of flow data files including DSPLAY, DSSUTL, MATHPK. Use MATHPK to compute adjusted local flows from incremental inflows and depletions (both regular interval and monthly constant). Use DSSUTL to obtain both a new and condensed catalogs. Use DSPLAY to plot inflows. (Hurst, Hayes)

Day 2
Tue, December 3

Preparation:	Already make a run of HEC-PRM for "base" conditions.
8:00 - 9:15	Explanation of penalty function concepts and method of derivation for MRD. Describe use of Lotus for computing penalty functions and generating .PRN file for use as input to DSSPD. (Moser)
9:30 - 11:45	Continue previous lecture. Hands on editing of penalty functions for Run2 (modified navigation penalty function at Sioux City) using Lotus. Compute composite curve using either Lotus or MATHPK. Derive edited, convex, composite penalty function using graphical editing capability of DSPLAY. (Moser, Hurst)
1:00 - 2:00	Description of HEC-PRM ASCII data input and the use of COED from MENUPRM to enter / edit the file. (Carl)
2:00 - 2:30	Hands-On Input for Run2 using COED from MENUPRM. Modify ASCII input data to bring in new penalty function. Make run of HEC-PRM for Run2. (Carl, Hurst)
2:45 - 3:15	Overview of post-processing using PRMPOST and MATHPK from MENUPRM in both automatic and "manual" modes. Description of the output DSS file from MATHPK. Brief description of MATHPK macro files. (Carl)
3:15 - 4:30	Hands-On post-processing. Get MRD formatted output and MATHPK tabular output. View results computed by MATHPK using DSPLAY. Compare results from the "base" run with those of "Run2". (Carl, Hurst)

Day 3
Wed, December 4

8:00 - 9:30	Review Procedures from yesterday. Evaluate and discuss results of altered Plan. (Carl, Moser, Hayes, Hurst)
9:30 - 9:45	Break
9:45 - 11:00	Operation Rules Development from HEC-PRM. (Lund)
11:00 - 12:00	Discussion / Conclusions (Burnham, et all)

Appendix F

HEC-PRM Progam Description

HEC-PRM

Hydrologic Engineering Center's Prescriptive Reservoir Model

Program Description

December 1991

Hydrologic Engineering Center
Water Resources Support Center
U.S. Army Corps of Engineers
609 Second Street
Davis, California 95616-4687

(916) 756-1104

Table of Contents

List of Figures

List of Tables

HEC-PRM

Hydrologic Engineering Center's Prescriptive Reservoir Model

Program Description

December 1991

Overview of HEC-PRM Procedures and Related Programs for MRD

The overall schematic of HEC-PRM and its relationship to other programs is shown in Figure 24. HEC-PRM requires the use of HEC's Data Storage System (HECDSS). HEC-PRM must read the penalty functions, local incremental inflows, and evaporation rates from HECDSS data files --- it cannot read them from any other source. As a result, several programs are used to enter and edit input data in preparation to applying HEC-PRM.

Typical Procedures for MRD

Although some operations may be performed in a different order, the following procedures are typical for the MRD study:

(1) Enter regular interval time series data in the HECDSS data file "MRD.DSS".

The regular interval time series data consists of incremental local inflows, incremental local depletions, and evaporation rates. The program "RDATA0" reads the file "D0DATA" which is normally input data for the MRD model. RDATA0 internally contains the node locations and corresponding pathname parts. Its also contains the distribution of evaporation. The file "D0DATA" contains monthly interval inflows and depletions which are stored in the HECDSS data file using separate pathnames. The file "D0DATA" also contains yearly evaporation (in feet) at the six mainstem reservoirs. RDATA0 converts these to monthly regular interval time series data using the internally stored monthly distribution. In addition, monthly varying depletions, entered in the MRD input data file "G1", may be stored as paired data using the DSSPD program.

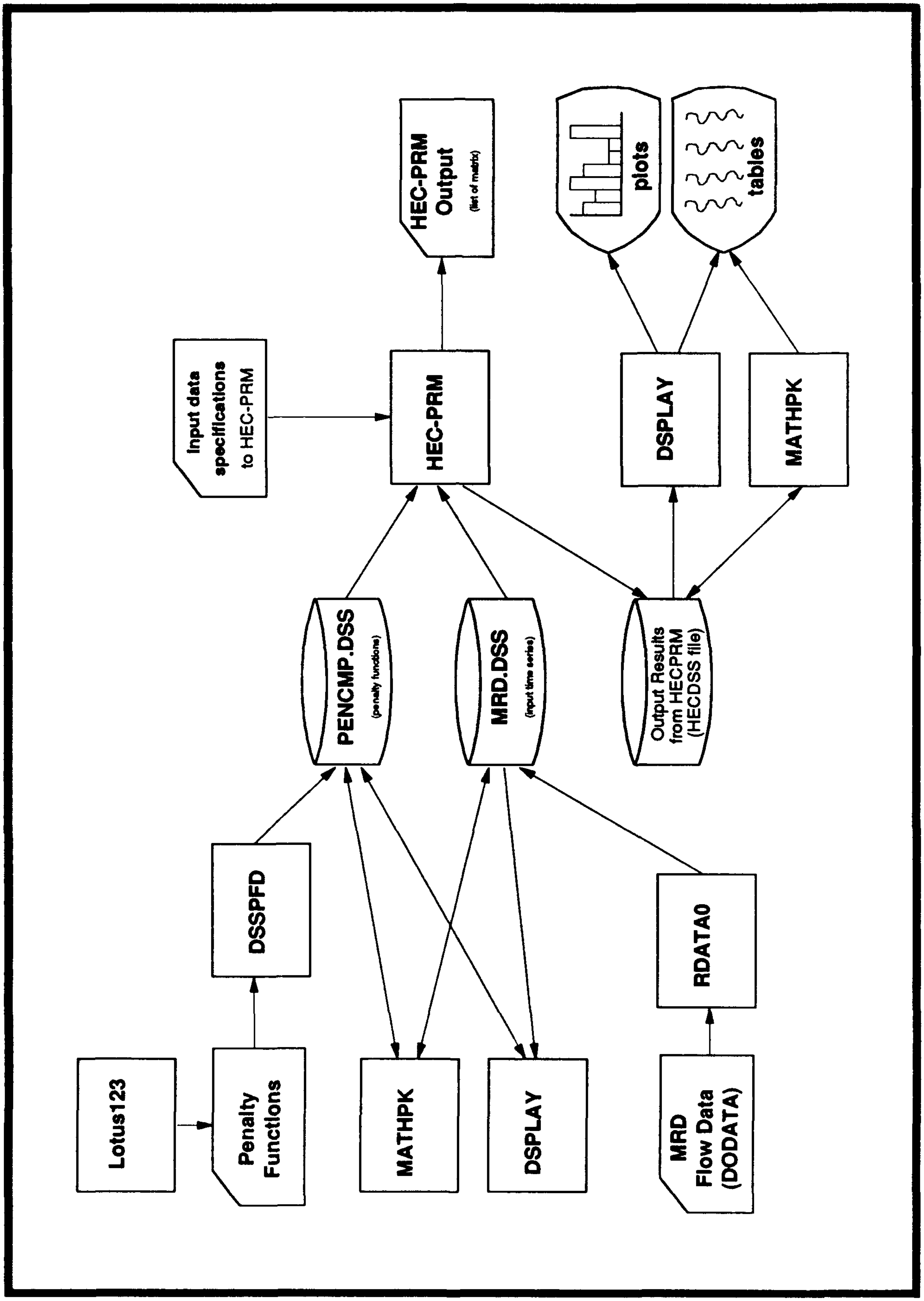
(2) Compute the adjusted local incremental inflow using MATHPK by adding inflows and depletions.

MATHPK allows the user to add, subtract, multiply, divide, etc. hydrographs and store the results in the same or a different HECDSS data file.

(3) Enter data and calculate the storage and flow penalty functions for each location and category (water supply, hydropower, etc).

These calculations were performed using Lotus 123, version 3.0. This step was performed by the Institute for Water Resources, U.S. Army Corps of Engineers.

Figure 25: Schematic of HECPRM and Related Programs



(4) Re-format and relocate penalty functions within the Lotus123 spreadsheets.

Using Lotus 123, version 3.0, an additional "sheet" was appended to each Lotus123 file. The penalty functions were copied to this appended sheet and were formatted in preparation for entry into a HECDSS data file.
The HECDSS pathname and appropriate header information are added before each penalty function (requiring additional rows for each penalty function). This section of the spreadsheet is "Print"ed to the ASCII file "xxxxxx.PRN" which corresponds to the worksheet "xxxxxx.WK3".

(5) Store the penalty functions in the HECDSS data file "PENCMP.DSS".

The program "DSSPD" reads the "xxxxxx.PRN" file from the Lotus123 spreadsheet and stores the penalty functions as "Paired Data Convention". Alternatively, the "Store DATAFILE" option of the PIP program may be used.

(6) Convert the penalty functions into "standard" units using MATHPK.

Conversion of penalty functions into "standard" units requires several types of manipulations. Storage penalties expressed as penalty (in millions of dollars) versus pool elevation (in feet) must be converted to penalty (in thousands of dollars) versus storage (in thousands of acre-feet). The MATHPK function "TABLE" facilitates this. Flow penalties expressed as penalty versus flow (in thousands of cubic feet per second) must be converted to penalty versus flow (in thousands of acre-feet per month). An average value of 30.5 days per month was used to determine this factor.

(7) Compute a composite function for each month at each location.

The ultimate goal is to have a separate pathname for each composite penalty function for each location. The original functions were a mix in which one category (e.g. hydropower) has one function which applies to all months of the year whereas another category (e.g. recreation) has separate functions for each month of the year. The original penalty functions which varied by month were stored using one pathname. The final "edited" penalty functions (those which the model uses for computations) must be stored in a separate pathname for each month for each location. This step created the "computed composite" function by adding functions for all categories for a given link. For example, to compute the June composite reservoir storage penalty function, the analyst must add the single hydropower capacity storage function, the June pool recreation function, and the June water supply function and store the result in the HECDSS data file.

(8) Estimate the convex "edited composite" penalty function from the "computed composite" penalty function.

Estimating the "edited composite" (or "model") penalty function accomplishes two goals: (1) Forming a convex function, and (2) Reducing the number of arcs describing the functions. The "computed composite" function contains many ordinates. The analyst must determine the simplest "edited" function that still adequately describes the "computed" function while remaining convex. The HECDSS-DSPLAY program is invoked to plot the "computed composite" function. The analyst then issues the "ED.D" command to add the additional simple "edited" function and save it in the HECDSS data file "PENCMP.DSS" with an appropriate unique pathname.

(9) Store Pertinent Paired Data Functions In The File PENCMP.DSS

Other paired data needs to be entered in the PENCMP.DSS DSS data file. This includes hydropower coefficients (A, B, and capacity), and duration analysis function (class intervals). Hydropower functions are entered using DSSPD and class intervals may be entered with DSSPD or computed and stored using MATHPK.

(10) Prepare HEC-PRM input data specifications.

The analyst invokes an editor (such as COED) to create and enter information pertinent to the HEC-PRM program. This is an ASCII file which contains some miscellaneous parameters such as the time window associated with the calculations, the HECDSS pathname parts under which the computed results are stored, factors which are applied against flow and penalty data, etc. The primary content of this file is a list of nodes and a list of links with associated information such as the connected nodes, the pathname parts for the penalty functions and the regular time series data (inflow and evaporation), upper and lower bounds, etc.

(11) Perform the network flow optimization.

To optimize the network, HEC-PRM reads the ASCII data specifications file, retrieves the appropriate time series data from the file "MRD.DSS", retrieves the appropriate "edited composite" penalty functions from the file "PENCMP.DSS", generates the network flow solver matrix, calls the solver, and stores the time series results in the output HECDSS data file.

(12) Display results graphically or tabularly.

The analyst may use the HECDSS-DSPLAY program to plot or tabulate the time series results. These include reservoir storage, reservoir releases, and channel flows. All flow and storage data is expressed in units of thousands of acre-feet. The MATHPK program may be used to convert these into other units such as pool storage in terms of elevation in feet or flow in terms of thousands of cubic feet per second. Extensive MATHPK macros are written to compute hydropower, duration functions, time series penalties, etc.

(13) Review the solver Matrix.

Figure 24 also depicts HEC-PRM writing output data to an ASCII file. It contains an echo of the user input including pathnames, warning messages, and a list of the solver matrix both before and after solution. While some of the information is useful, it is very painfull to look at the solver matrix and should be done only for short optimizations (a limited number of arcs). Usefull output is created using the HECDSS-DSPLAY program and possibly MATHPK. These are both generalized programs which represent 2 of the many HECDSS utility programs. HECDSS-DSPLAY facilitates the output of graphs in the standard HECDSS graphics format and includes the capability of creating graphics metafiles which may be imported to other software packages such as word processors and drawing programs.

General Description of HEC-PRM Software

HEC-PRM consists of about 40 program specific subroutines of which 8 are the generalized solver. It also utilizes many routines from HEC's software library including HECDSS routines. HEC-PRM's routines do the following:

- assign disk files.
- read user defined input.
- print user input.
- read all penalty functions (storage and flow) and all time series data (evaporation and inflow) from input HECDSS data files.
- generate the solver matrix.
- print the matrix.
- call the solver routines.
- print the computed solver matrix.
- store the results in an output DSS data file.
- close all disk files.

Structure of HEC-PRM

The internal flow of information within HEC-PRM is shown in Figure 25. HEC-PRM retrieves input data from three sources:

(1) An ASCII data specifications file,
(2) An HECDSS data file containing penalty functions, and
(3) An HECDSS data file containing regular interval time series data.

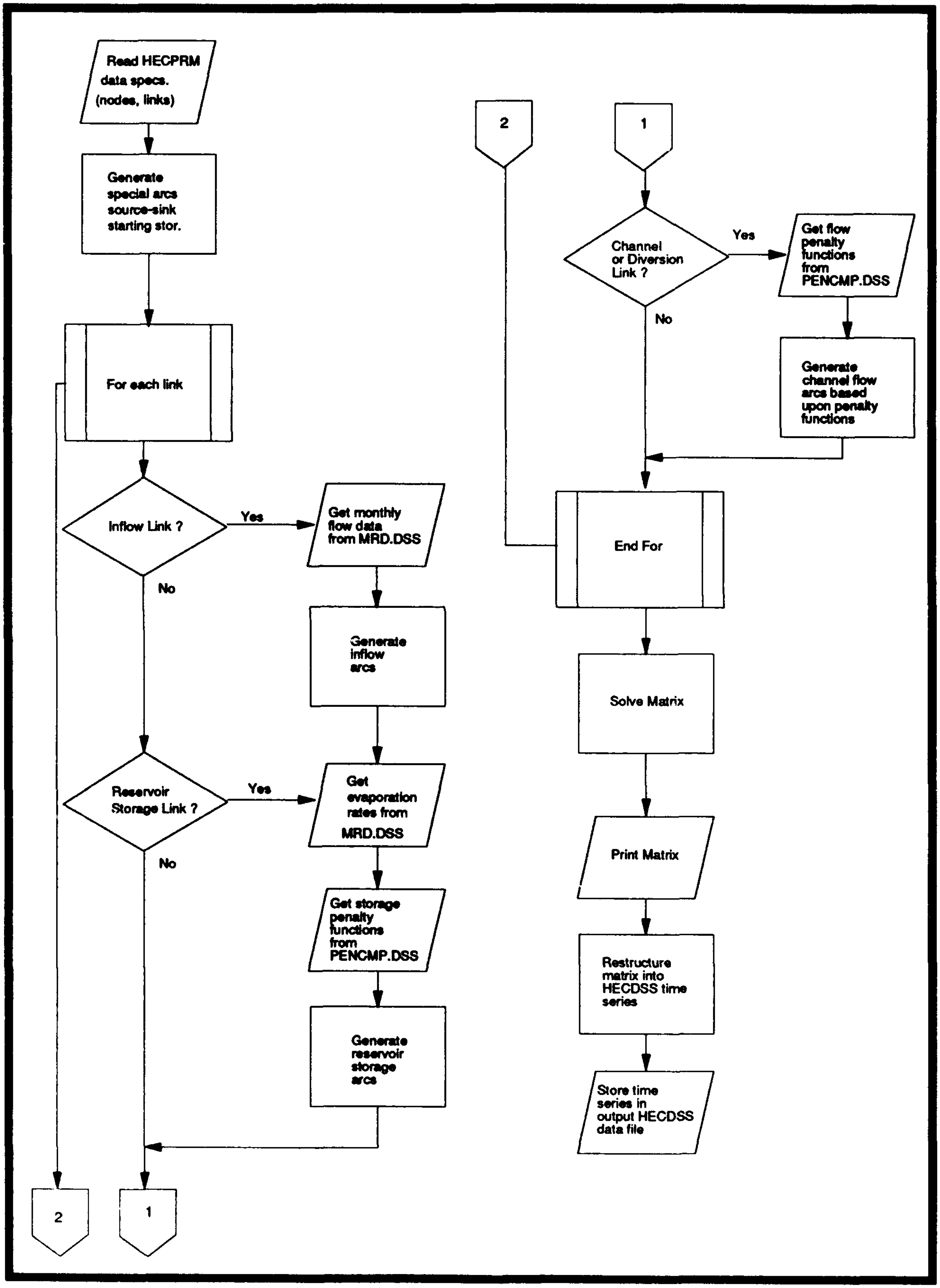
The input data relationships are shown in Figure 24. HEC-PRM reads the ASCII data specifications file first. It defines many miscellaneous parameters such as the computational time window, the nodes, and the links. HEC-PRM stores the number of nodes and links as well as associated information such as the nodes which are connected by each link and the pathnames which define the storage location within the HECDSS data files for the penalty functions, inflows, and evaporation rates. It then generates the solver matrix which consists of arcs each of which is defined by the following the parameters:

(1) Source node,
(2) Target node,
(3) Lower bound,
(4) Upper bound,
(5) Unit cost,
(6) Amplitude (for the gains solver), and
(7) Flow (initialized to zero).

There are several other scalar parameters that must also be set. HEC-PRM builds this matrix by generating 1 or more "special" arcs and then processing each link in the same sequence in which they were entered by the user. The first arc is always the arc between the super source and the super sink. The second through NRES+1 arcs contain the starting storage for each reservoir where "NRES" is the number of reservoirs. The subsequent arcs are dependent upon the order and type of links entered by the user. When appropriate, data is retrieved from HECDSS data files. For example, the inflow links require the retrieval of local incremental adjusted inflow, storage links require the retrieval of evaporation rates and storage penalty functions, etc.

If the user has defined constraints for a given link, the lower and upper bounds on the arcs are set accordingly. The number of arcs associated with a

Figure 26: Internal Processes of HECPRM



given link for a given month is exactly the same as the number of line segments in the penalty function. All inflow arcs have one arc per time period and have lower and upper bounds which vary for every time period and are set to the local inflow with a unit cost of zero. Once the solver matrix is filled and appropriate parameters are set (such as the number of arcs, the number of nodes, and the total system inflow), HEC-PRM calls the network flow with gains solver. HEC-PRM tracks the progress of solution by displaying on the computer screen the number of iterations and the computed total flow. The solver continues to iterate until a least cost solution is determined. The typical application contains many unknowns with far fewer equations. Therefore, there can be many solutions to the problem. Minor changes to input data can cause the solver to compute an entirely different solution.

Once the solution is determined, the solver fills the "FLOW" array with the computed flow in each arc of the matrix. HEC-PRM can then "post-process" this information by adding the flow in all arcs for each time period and storing monthly regular interval time series data in the output HECDSS data file. The output monthly time series data includes not only flow and storage but also the total cost by month for each link. The total cost is computed using the "edited composite convex" penalty function defined by the user and retrieved by HEC-PRM from the HECDSS data file "PENCMP.DSS".

Presently, HEC-PRM also generates an integer matrix which is written to the file "SUPERK.SK". This matrix is formated so that it can be retrieved by an alternative solver "SUPERK" which does not contain any "gains" capability and is therefore not appropriate for the MRD study. Also, at this time, HEC-PRM cannot post-process the SUPERK matrix and store the results in an HECDSS data file.

HEC-PRM can also generate data for use with the NETG solver (a commercially available solver). The results from this solver are post-processed outside of the HEC-PRM program.

Executing HEC-PRM

HEC-PRM requires the use of the HECDSS software. All input penalty functions are read from one DSS data file, all time series input data is read from another DSS data file, and the "understandable" results are written to a third HECDSS data file. The files may be defined using "MENUPRM". Alternatively, these data files may be defined at the DOS prompt which executes HEC-PRM as shown below:

```
HECPRM: Prescriptive Reservoir Model - Vers. December 1, 1991
   UNIT        KEYWORD              *ABREV       **MAX    DEFAULT
      5        INPUT                 I            64      CON
      6        OUTPUT                O            64      CON
     95        F95                   F            64      SCRATCH.035
    NOP        TS_DSS_IN             T            64      SCRATCH.031
    NOP        PF_DSS_IN             P            64      SCRATCH.032
    NOP        DSSOUT                D            64      SCRATCH.033
      1        MSG                   M            64      HECPRM.ERR
     29        TRACE                 TR           64      SCRATCH.009

   * ABREV - SHORTEST ABBREVIATION ALLOWED FOR KEYWORD
  ** MAX - MAXIMUM # OF CHARACTERS FOR FILENAME (OR STRING)
```

To execute the program, you must enter the program name followed by the appropriate file names. Use the "*ABREV" codes to define the file type. The following files should be defined: INPUT, OUTPUT, TS_DSS_IN, PF_DSS_IN, and DSSOUT. The file "HECPRM.ERR" contains error messages. It is supplied with the program and you should copy it into your local subdirectory (where the data is located). An example command to execute HEC-PRM is as follows:

```
HECPRM I=I5Y0D.PRI O=I5Y0D.PRO T=MRD.DSS P=PENCMP.DSS D=I5Y0D.DSS
```

The corresponding files are:

INPUT I5Y0D.PRI
User input data which defines many items including the time window, the nodes, the links, the pathname parts for the penalty functions, the pathname parts for the time series data, etc.

OUTPUT I5Y0D.PRO
The tabular output from the program including a listing of the user input (file I5Y0D.PRI) and a listing of the solver matrix before and after the solution.

TS_DSS_IN MRD.DSS
The HECDSS data file which contains all input time series data including the rate of evaporation (EV records) and incremental inflows (IN records).

PF_DSS_IN PENCMP.DSS
The HECDSS data file which contains all input penalty functions including flow (PQ records) and reservoir storage (PS records).

DSSOUT I5Y0D.DSS
The HECDSS data file which contains all computed time series data. The local inflows are also written to this file.

Data Units

All flow data are entered in thousands of acre-feet (KAF). All penalties are entered in thousands of dollars (K$). All evaporation rates are entered in feet per month (FT). The data may be entered in other units (e.g. penalties in thousands of dollars). However, all flow and storage must be entered in consistent units (KAF) for both the time series data as well as penalty functions. All penalties must be in consistent units (K$). You may enter factors on the J1 record to change the magnitude of the data for the solver. For example, all unit costs in thousands of dollars could be multiplied by 0.001 so that the solver would use unit costs in millions of dollars.

Format of HECDSS data

The HECDSS software has several standard data conventions. These are useful because many programs recognize these conventions and can easily manipulate and display data which meets these standard conventions. Time series data is stored using the regular interval time series data convention. Penalty functions are stored using the paired function data convention.

For the MRD study, two programs were developed to read data from ASCII files supplied by MRD. One program reads the standard monthly flow data files and stores the data in a HECDSS data file. The other program reads output from the "V1.EXE" program and reformats it so that the program HECDSS-DSSRTS can be used to store it in a HECDSS data file.

Penalty functions and rating curves may be entered with the standard programs HECDSS-PIP or HECDSS-DSSPD. For the MRD study, this data was entered in Lotus spreadsheets, appropriate additional information was added, and a Lotus ".PRN" file was created. This ".PRN" file is compatible with the input file used by the HECDSS-DSSPD program. The HECDSS-DSSPD can be used to read this file and store it in a HECDSS data file. All penalty functions are defined with the storage (or flow or elevation) variable first and the penalty variable second. The HECDSS-MATHPK program is used to convert penalty functions into proper units (e.g. convert elevation to storage or convert flow in CFS to volume in KAF/MONTH).

Order of User Input

The records described on the following pages should be entered in the following order:

First	All job control records such as "TIME", "J1", "IDENT", and "ZW".
Second	All node identifier records ("NODE").
Third	All link definition records including "LINK", "BL", "BU", "PS", "PQ", "EV", and "IN". The "LINK" record is the first record of each link. There is one "LINK" record for each link in the network. All other "LINK" records which are required to define that link follow the "LINK" record.

Comment records ("..", "**", or " ") may be entered anywhere in the input data stream.

Note:

In the following description, the character "ƀ" indicates a blank character.

Input Data Records

..

Columns	Variable	Value	Description
1-2	CINREC	..	Comment Record. Columns 3-n are printed in output. This record may be entered anywhere within the input data stream. There is no limit to the number of comment records.

**

Columns	Variable	Value	Description
1-2	CINREC	**	Comment Record. Columns 3-n are printed in output. This record may be entered anywhere within the input data stream. There is no limit to the number of comment records.

b b

Columns	Variable	Value	Description
1-2	CINREC	(blank)	Comment Record. Columns 3-n are printed in output. This record may be entered anywhere within the input data stream. There is no limit to the number of comment records.

IDENT, TIME

IDENT

Columns	Variable	Value	Description
1-5	CINREC	IDENT	This record contains the code names for the Super Source and Super Sink nodes. It is an optional record. If you wish to use the default code names of "S_SOURCE" and "S_SINK", you need not enter this record. If entered, it must precede all other records which reference these nodes such as the "NODE " and "LINK" records.
11-20	IDNODE(1)	character	The character identification for the Super Source node. Default is "S_SOURCE". (Maximum 10 characters)
21-30	IDNODE(2)	character	The character identification for the Super Sink node. Default is "S_SINK". (Maximum 10 characters)

TIME

Columns	Variable	Value	Description
1-4	CINREC	TIME	This record contains the time window for which calculations are performed. It is a required record.
11-80	CINREC	character	Contains the time window in free-format. There must be a starting date and ending date. The starting date must be entered first followed by the ending date. The date consists of the 3 character month and the integer year without any space separating them. The day is not entered as part of the time window since all computations are done on a monthly time interval. The year may be either 2 or 4 digit year (the 2 digit year is assumed to be within the years 1900-1999). An example entry is: TI JAN1930 DEC1940 Internally, the computation time window is stored in variables: YRWIN(2) Year MONWIN(2) Month DAYWIN(2) Day

J1

Columns	Variable	Value	Description
1-2	CINREC	J1	This record contains specifications for this simulation. The specifications include network solver constants, function factors, and data units.
3-10	EPS	+ real no.	EPS is a small positive number which is used by the Jensen-Bhaumik "Network with Gains" Solver. HEC-PRM uses EPS to test for iteration convergence and to determine the least cost flow path. (Default is 1.0E-05; Must be a positive number).
11-20	BIG	+ real no.	BIG is a large positive number which is used by the Jensen-Barnes "Network with Gains" Solver. HEC-PRM uses BIG in several areas including the upper flow bound for links which have no specific physical boundary. An example is a channel link between two nodes. (Default is +1.0E+06; Must be a positive number).
21-30	FACFLO	blank, + real no.	FACFLO scales all incremental inflows for calculations by the network solver. HEC-PRM multiples all inflows by FACFLO before calling the solver and then divides the computed flows by FACFLO after the solution and before storing results in the output DSS datafile. (Default is 1.0; Must be a positive number).
31-40	FACCST	blank, + real no.	FACCST scales all unit costs which are derived from the penalty functions. The scaled costs are used by the network solver. Each arc contains a unit cost. HEC-PRM multiples all unit costs by FACCST before calling the solver and then divides the unit costs by FACCST after the solution and before storing cost time series data in the output DSS datafile. (Default is 1.0; Must be a positive number).

ZW

Columns	Variable	Value	Description
1-2	CINREC	ZW	This record contains the character identification for pathname part F of the results for this simulation.
3-80	SIMLID	character	SIMLID is the portion of pathname part F which is used to identify this simulation. It is used to store results under separate DSS records for different simulations (e.g. to differentiate results after changing penalty functions between simulations). HEC-PRM uses SIMLID to form pathname part F. (Maximum of 20 characters; recommend using less).

NODEbbbb

Columns	Variable	Value	Description
1-8	CINREC	NODEbbbb	These records contain the identification code for each node and the associated character description. These are required records. All "NODE " records must be entered before entering any "LINK" records. There must be one "NODE " record for each node in the network. HEC-PRM automatically defines the first two nodes to be the Super Source and the Super Sink. You may optionally define the Super Source and Sink on "NODE " records. If you include the "IDENT" record, it must precede the "NODE " records.
11-20	IDNODE	character	The code identifier for this node. This node identifier is used on the "LINK" records to define the network. Each IDNODE must be unique for this input data set.
21-30	RSTO1	blank, + real no.	The initial storage at the beginning of the simulation for this node if it is a reservoir. If this node is not a reservoir, leave this field blank. HEC-PRM defines this node as a reservoir if this field is non-blank. RSTO1 is entered in thousands of acre-feet. (Default is blank; if reservoir, must be a positive real number).
31-40	RSTOK	blank, + real no.	The factor for converting reservoir capacity into area for use in computing evaporation. It is computed by estimating a representative slope of the area-capacity curve. The numerator is the change in area (in thousands of acres) and the denominator is the change in volume (in thousands of acre-feet). (Default is 0.1; Must be zero or a positive real number).
41-50	RSTOT	blank, + real no.	The target reservoir storage for the last time period of the optimization. Enter RSTOT only for reservoirs.

NODE

LINK

Columns	Variable	Value	Description
1-4	CINREC	LINK	These records contain the information for the links between nodes. They are required records. HEC-PRM defines the overflow arc between the Super Source and the Super Sink to be the first arc in the network. You need not enter a "LINK" record to define this arc but you may. The typical reason why you might enter this link is to specify the unit cost.
11-20	LINKTY	character	The type of link. (No default, the user must enter a value). Must match one of the following codes: Code — Description CHAN — Channel link. DIVR — Diversion link (includes link from the Super Source to the Super Sink and the link from the last node to the Super Sink). INFL — Inflow link. RREL — Reservoir release link. RSTO — Reservoir storage link. HREL — Hydropower release link.
21-30	LIFROM	character	The node identification from which flow travels through this link. It must match one of the identifiers entered on the "NODE" records or the Super Source or the Super Sink. (No default, user must enter node identifier in this field; Maximum of 10 characters)
31-40	LINKTO	character	The node identification to which flow travels through this link. It must match one of the identifiers entered on the "NODE" records or the Super Source or the Super Sink. (No default, user must enter node identifier in this field; Maximum of 10 characters)
41-50	LIAMP	+ real no.	The amplitude to be used for this link. If left blank, it is set to "1.0". (Default is "1.0"; Must be a positive number)
51-60	LICOST	real no.	The constant cost for all time periods for this link. This field is normally left blank. If a number is entered here, do not enter penalty functions. (Default is "0.0"; May be positive, negative, or zero)
61-70	LILOWR	+ real no.	The lower bound for this link. This is the lowest allowable value of volume (flow / storage) in the penalty function for which the algorithm will solve. This is normally left blank. Care must be exercised in using bounds because the algorithm

will abort if it cannot determine a feasible solution. (Default is "0.0"; Must be zero or a positive number)

LINK (continued)

Columns	Variable	Value	Description
71-80	LIUPPR	+ real no.	The upper bound for this link. This is the highest allowable value of volume (flow / storage) in the penalty function for which the algorithm will solve. This is normally left blank. Care must be exercised in using bounds because the algorithm will abort if it cannot determine a feasible solution. (Default is to set LIUPPR equal to parameter "BIG"; Must be positive number)

LD

Columns	Variable	Value	Description
1-8	CINREC	LD	This record contains the description for the current link defined on the "LINK" record. It is an optional record.
11-n	LIDESC	character	The character description for this node. It can be any textual information.

BL, BU

BL

Columns	Variable	Value	Description
1-2	CINREC	BL	This record contains information about the lower bounds of the penalty function. If the lower bound remains constant throughout the simulation, it is entered on the "LINK" record. If it follows a monthly pattern (all January bounds are the same, all February bounds are the same but different than January, etc.), then it must be entered on this record.
3-n	LILOWR	0, + real no.	The monthly varying lower bounds is entered in free-format. Twelve values must be entered. The first value corresponds to January. If you enter adjacent commas, the corresponding value is defined by the value entered on the LINK record.

BU

Columns	Variable	Value	Description
1-2	CINREC	BU	This record contains information about the upper bounds of the penalty function. If the upper bound remains constant throughout the simulation, it is entered on the "LINK" record. If it follows a monthly pattern (all January bounds are the same, all February bounds are the same but different than January, etc.), then it must be entered on this record.
3-n	LIUPPR	0, + real no.	The monthly varying upper bounds is entered in free-format. Twelve values must be entered. The first value corresponds to January. If you enter adjacent commas, the corresponding value is defined by the value entered on the LINK record.

EV

Columns	Variable	Value	Description
1-2	CINREC	EV	This record contains HECDSS pathname parts and instructs the HEC-PRM to read time series data for evaporation. The evaporation should be expressed as a rate per month in feet. This record must be entered only for Reservoir Storage ("RSTO") links.
3-N	CINREC	character	The pathname parts are entered in free-format. The pathname part is preceded by an equal "=" sign which is preceded by the part identifier (A through F). Parts D and E are assumed. Part E is the time interval (1MON), and part D is the standard block time for regular interval time series data. Part D is generated from the time window entered on the "TIME" record. If part C is not entered, the HEC-PRM assumes it will be "EVAP_RATE". If a part is blank, enter the part identifier, the equal sign, and at least one blank. The HEC-PRM remembers time series parts from the last entry (either "IN" or "EV" records) --- you need only update necessary parts. Example entry: EV B=FTPK

IN

Columns	Variable	Value	Description
1-2	CINREC	IN	This record contains HECDSS pathname parts and instructs the HEC-PRM to read time series data for inflow. The flow should be expressed as a volume per month in 1,000 Acre-Feet. This record may be entered for any inflow link.
3-N	CINREC	character	The pathname parts are entered in free-format. The pathname part is preceded by an equal "=" sign which is preceded by the part identifier (A through F). Parts D and E are assumed. Part E is the time interval (1MON), and part D is the standard block time for regular interval time series data. Part D is generated from the time window entered on the "TIME" record. If part C is not entered, the HEC-PRM assumes it will be "FLOW". If a part is blank, enter the part identifier, the equal sign, and at least one blank. The HEC-PRM remembers time series parts from the last entry (either "IN" or "EV" records) --- you need only update necessary parts. Example entry: IN A= B=FTPK F=

PS

Columns	Variable	Value	Description
1-2	CINREC	PS	This record contains HECDSS pathname parts and instructs the HEC-PRM to read paired function reservoir storage penalty functions from the HECDSS data file. The penalty should be expressed in thousands of dollars and the storage in 1,000 Acre-Feet. It also identifies the month or range of months that this penalty function is applicable. There can be a separate penalty function for each of the months within a year. The penalty function cannot vary from year-to-year. The first penalty function must be for January or for a range of consecutive months starting in January. The functions must be in chronological order (e.g. you cannot enter the August function before the July function). You must define a function for each month.
3-N	CINREC	character	The pathname parts are entered in free-format. The pathname part is preceded by an equal "=" sign which is preceded by the part identifier (A through F). The months are entered in free-format using the 3 character month identifier (e.g. "FEB"). If the penalty function spans 2 or more consecutive months, the starting and ending months must be separated by the "-" (dash) character. Similar to the pathname parts, the months are preceded by an "=" (equal) sign which is preceded by the characters "MO". If part C is not entered, the HEC-PRM assumes it will be "STOR-PNLTY_EDT". If a part is blank, enter the part identifier, the equal sign, and at least one blank. The HEC-PRM remembers paired function parts from the last entry (either "PS" or "PQ" records) --- you need only update necessary parts. Example entries:

```
PS  MO=JAN-MAR  B=FTPK E=JAN
...
PS MO=JUL B=GARR E=JUL
...
PS MO=JAN-DEC B=OAHE
```

PQ

Columns	Variable	Value	Description
1-2	CINREC	PQ	This record contains HECDSS pathname parts and instructs the HEC-PRM to read paired function flow penalty functions from the HECDSS data file. This record is used with the following types of LINK records: CHAN, RREL, HREL, and DIVR. The penalty should be expressed in thousands of dollars and the flow in 1,000 Acre-Feet. It also identifies the month or range of months for which this penalty function is applicable. There can be a separate penalty function for each of the months within a year. The penalty function cannot vary from year-to-year. The first penalty function must be for January or for a range of consecutive months starting in January. The functions must be in chronological order (e.g. you cannot enter the August function before the July function). You must define a function for each month.
3-N	CINREC	character	The pathname parts are entered in free-format. The pathname part is preceded by an equal "=" sign which is preceded by the part identifier (A through F).

The months are entered in free-format using the 3 character month identifier (e.g. "FEB"). If the penalty function spans 2 or more consecutive months, the starting and ending months must be separated by the "-" (dash) character. Similar to the pathname parts, the months are preceded by an "=" (equal) sign which is preceded by the characters "MO".

If part C is not entered, the HEC-PRM assumes it will be "FLOW-PNLTY_EDT".

If a part is blank, enter the part identifier, the equal sign, and at least one blank. The HEC-PRM remembers paired function parts from the last entry (either "PS" or "PQ" records) --- you need only update necessary parts.

Example entries:

```
PQ  MO=JAN-MAR A=  B=FTPK-GARR E=JAN F=
...
PQ MO=JUL B=GARR-OAHE E=JUL
...
PQ MO=JAN-DEC B=OAHE-BEND
...
PQ MO=JAN B=FTPK-GARR C=FLOW-PNLTY_HPE_EDT
```

Terminate Input - Introduction

Any one of the following commands may be entered. If more than one command is entered, only the first is observed. The entry of one of these commands instructs HEC-PRM to terminate reading user input, generate the network matrix, call the solver, and store the results in the output DSS data file.

STOP

Columns	Variable	Value	Description
1-4	CINREC	STOP	Terminate user input and commence generating and solving the network.

FINISH

Columns	Variable	Value	Description
1-4	CINREC	STOP	Terminate user input and commence generating and solving the network.

QUIT

Columns	Variable	Value	Description
1-4	CINREC	STOP	Terminate user input and commence generating and solving the network.

Terminate Input

Example User Input and Output

HEC-PRM

Hydrologic Engineering Center's Prescriptive Reservoir Model

Example User Input and Output

December 1991

The following lists a typical HEC-PRM output data set. The input data records are listed as the first part of the output data. This particular data set is for the 1 year time window from March 1965 through February 1966. By modifying the "TIME" record, it could easily be adapted to the 5 year validation period or the 23 year critical period.

{Banner Page}

```
                    Reservoir System Operation Optimization
                        Version  0.2.01;  October 1991
                       IBM-PC Compatible  (Lahey 32bit)
                       Run date 26NOV91   time 17:41:05


                                  HECPRM


                        U.S. Army Corps of Engineers
                        Hydrologic Engineering Center
                         609 Second Street, Suite B
                           Davis, California 95616
                              (916) 756-1104
```

{Open DSS files and echo Input Data}

```
    -----DSS---ZOPEN:  Existing File Opened,  File: TSIN.DSS
                       Unit:  71;  DSS Version: 6-GO
    -----DSS---ZOPEN:  Existing File Opened,  File: PENCMP.DSS
                       Unit:  72;  DSS Version: 6-GO
    -----DSS---ZOPEN:  Existing File Opened,  File: EXMPL1.DSS
                       Unit:  73;  DSS Version: 6-GO
..        Test EXMPL1
..        1 Year period
..        Evaporation included
..        Reservoir bounds included
..        HEC-PRM Picks ending storage
..        Set bypass link cost to zero
..
ZW        F=EXMPL1
..
IDENT     S_SOURCE  S_SINK
..
TIME      MAR65 FEB66
J1 1.0E-05    1.0E+06        1.0    1.0E+03
               EPS            BIG          FACFLO          FACCST
         0.0000100       1000000.        1.00000      1000.00000
    0.1000000E-04  0.1000000E+07  0.1000000E+01  0.1000000E+04
..
NODE      FTPK           13900.       .011
ND        Fort Peck
NODE      FTPKR
ND        Fort Peck
NODE      GARR           15605.       .013
ND        Garrison
NODE      GARRR
ND        Garrison
NODE      OAHE           17739.       .012
ND        Oahe
NODE      OAHER
ND        Oahe
NODE      BEND            1730.       .024
ND        Big Bend
NODE      BENDR
```

```
ND       Big Bend
NODE     FTRA           3473.      .013
ND       Fort Randall
NODE     FTRAR
ND       Fort Randall
NODE     GAPT            357.      .033
ND       Gavins Point
NODE     GAPTR
ND       Gavins Point
NODE     SUX
ND       Sioux City
NODE     OMA
ND       Omaha
NODE     NCNE
ND       Nebraska City
NODE     MKC
ND       Kansas City
NODE     BNMO
ND       Boonville
NODE     HEMO
ND       Hermann
NODE     STL
ND       St. Louis
..
LINK     DIVR      S_SOURCE  S_SINK          1.0       0.0
LD       Link to balance flow throughout computations
..
LINK     INFLOW    S_SOURCE  FTPK            1.0       0.0
LD       Inflow into Fort Peck Reservoir
IN       A=     B=FTPK C=FLOW_LOC E=1MON  F=
..
LINK     RSTOR     FTPK      FTPK                                4211     17714
LD       Storage in Fort Peck Reservoir
PS       MO=JAN-MAR A=     B=FTPK C=STOR-PNLTY_EDT D= E=JAN F=
PS       MO=APR-JUN C=STOR-PNLTY_EDT E=APR
PS       MO=JUL E=JUL
PS       MO=AUG E=APR
PS       MO=SEP E=SEP
PS       MO=OCT E=APR
PS       MO=NOV E=SEP
PS       MO=DEC E=DEC
EV       C=EVAP_RATE F=
..
LINK     HREL      FTPK      FTPKR
LD       Hydropower release from Ft. Peck
PQ       MO=JAN-DEC B=FTPK-GARR C=FLOW-PNLTY_HPE_EDT E=  F=
..
LINK     RREL      FTPKR     GARR
LD       Release from Fort Peck Reservoir to Garrison
PQ       MO=JAN-MAR B=FTPK-GARR C=FLOW-PNLTY_EDT E=JAN F=
PQ       MO=APR E=APR
PQ       MO=MAY-JUN E=MAY
PQ       MO=JUL E=JUL
PQ       MO=AUG E=AUG
PQ       MO=SEP E=SEP
PQ       MO=OCT E=OCT
PQ       MO=NOV-DEC E=JAN
..
LINK     INFLOW    S_SOURCE  GARR            1.0       0.0
LD       Inflow into Garrison Reservoir
IN       B=GARR F=
..
LINK     RSTOR     GARR      GARR                                4990     22430
LD       Storage in Garrison Reservoir
PS       MO=JAN-MAR B=GARR C=STOR-PNLTY_EDT E=JAN F=
PS       MO=APR E=APR
PS       MO=MAY E=MAY
PS       MO=JUN E=JUN
PS       MO=JUL E=JUL
PS       MO=AUG E=AUG
PS       MO=SEP E=SEP
PS       MO=OCT E=OCT
PS       MO=NOV E=NOV
PS       MO=DEC E=JAN
EV       C=EVAP_RATE F=
..
LINK     HREL      GARR      GARRR
LD       Release from Garrison Reservoir to Oahe (energy)
PQ       MO=JAN-DEC B=GARR-OAHE C=FLOW-PNLTY_HPE_EDT E=  F=
..
..
LINK     RREL      GARRR     OAHE
LD       Release from Garrison Reservoir to Oahe
PQ       MO=JAN-MAR B=GARR-OAHE C=FLOW-PNLTY_EDT E=JAN F=
PQ       MO=APR E=APR
PQ       MO=MAY-JUN E=MAY
PQ       MO=JUL E=JUL
PQ       MO=AUG E=AUG
PQ       MO=SEP E=SEP
PQ       MO=OCT E=OCT
PQ       MO=NOV-DEC E=[illegible]
..
```

```
LINK      INFLOW    S_SOURCE  OAHE           1.0       0.0
LD        Inflow into Oahe Reservoir
IN        B=OAHE F=
..
LINK      RSTOR     OAHE      OAHE                               5451      22240
LD        Storage in Oahe Reservoir
PS        MO=JAN-MAR B=OAHE C=STOR-PNLTY_EDT E=JAN F=
PS        MO=APR E=APR
PS        MO=MAY E=MAY
PS        MO=JUN-AUG E=JUN
PS        MO=SEP E=SEP
PS        MO=OCT E=OCT
PS        MO=NOV-DEC E=JAN
EV        C=EVAP_RATE F=
..
LINK      HREL      OAHE      OAHER
LD        Release from Oahe Reservoir to Big Bend Reservoir (energy)
PQ        MO=JAN-DEC B=OAHE-BEND C=FLOW-PNLTY_HPE_EDT E=  F=
..
LINK      RREL      OAHER     BEND
LD        Release from Oahe Reservoir to Big Bend Reservoir
PQ        MO=JAN-MAR B=OAHE-BEND C=FLOW-PNLTY_EDT E=JAN F=
PQ        MO=APR E=APR
PQ        MO=MAY-JUN E=MAY
PQ        MO=JUL E=JUL
PQ        MO=AUG E=AUG
PQ        MO=SEP E=SEP
PQ        MO=OCT E=OCT
PQ        MO=NOV-DEC E=JAN
..
LINK      RSTOR     BEND      BEND                               1696      1813
LD        Storage in Big Bend Reservoir
PS        MO=JAN-FEB B=BEND C=STOR-PNLTY_EDT E=JAN F=
PS        MO=MAR E=MAR
PS        MO=APR E=APR
PS        MO=MAY E=MAY
PS        MO=JUN E=APR
PS        MO=JUL E=JUL
PS        MO=AUG E=JUL
PS        MO=SEP E=SEP
PS        MO=OCT E=OCT
PS        MO=NOV E=MAR
PS        MO=DEC E=DEC
EV        C=EVAP_RATE F=
..
LINK      HREL      BEND      BENDR
LD        Release from Big Bend Reservoir to Fort Randall Reservoir (energy)
PQ        MO=JAN-DEC B=BEND-FTRA C=FLOW-PNLTY_HPE_EDT E=  F=
..
LINK      RREL      BENDR     FTRA
LD        Release from Big Bend Reservoir to Fort Randall Reservoir
PQ        MO=JAN-MAR B=BEND-FTRA C=FLOW-PNLTY_EDT E=JAN F=
PQ        MO=APR E=APR
PQ        MO=MAY-JUN E=MAY
PQ        MO=JUL E=JUL
PQ        MO=AUG E=AUG
PQ        MO=SEP E=SEP
PQ        MO=OCT E=OCT
PQ        MO=NOV-DEC E=JAN
..
LINK      INFLOW    S_SOURCE  FTRA           1.0       0.0
LD        Inflow into Fort Randall Reservoir
IN        B=FTRA F=
..
LINK      RSTOR     FTRA      FTRA                               1568      4589
LD        Storage in Fort Randall Reservoir
PS        MO=JAN-MAR B=FTRA C=STOR-PNLTY_EDT E=JAN F=
PS        MO=APR E=APR
PS        MO=MAY E=MAY
PS        MO=JUN E=JUN
PS        MO=JUL E=JUN
PS        MO=AUG E=AUG
PS        MO=SEP E=SEP
PS        MO=OCT E=MAY
PS        MO=NOV E=NOV
PS        MO=DEC E=JAN
EV        C=EVAP_RATE F=
..
LINK      HREL      FTRA      FTRAR
LD        Release from Fort Randal Reservoir to Gavins Point Reservoir (energy)
PQ        MO=JAN-DEC B=FTRA-GAPT C=FLOW-PNLTY_HPE_EDT E=  F=
..
..
LINK      RREL      FTRAR     GAPT
LD        Release from Fort Randal Reservoir to Gavins Point Reservoir
PQ        MO=JAN-MAR B=FTRA-GAPT C=FLOW-PNLTY_EDT E=JAN F=
PQ        MO=APR E=APR
PQ        MO=MAY-JUN E=JUN
PQ        MO=JUL E=JUL
PQ        MO=AUG E=AUG
PQ        [illegible]
PQ        [illegible]
```

```
PQ        MO=NOV-DEC E=JAN
..
LINK      INFLOW    S_SOURCE  GAPT
LD        Inflow into Gavins Point Reservoir
IN        B=GAPT F=
..
LINK      RSTOR     GAPT      GAPT                             340        432
LD        Storage in Gavins Point Reservoir
PS        MO=JAN-FEB B=GAPT C=STOR-PNLTY_EDT E=JAN F=
PS        MO=MAR E=MAR
PS        MO=APR E=APR
PS        MO=MAY-JUN E=MAY
PS        MO=JUL E=JUL
PS        MO=AUG E=AUG
PS        MO=SEP E=MAR
PS        MO=OCT-DEC E=JAN
EV        C=EVAP_RATE F=
..
LINK      HREL      GAPT      GAPTR
LD        Release from Gavins Point to Sioux City (energy)
PQ        MO=JAN-DEC B=GAPT-SUX C=FLOW-PNLTY_HPE_EDT E=  F=
..
LINK      RREL      GAPTR     SUX
LD        Release from Gavins Point to Sioux City
PQ        MO=JAN-MAR B=GAPT-SUX C=FLOW-PNLTY_EDT E=JAN F=
PQ        MO=APR E=APR
PQ        MO=MAY-JUN E=MAY
PQ        MO=JUL E=JUL
PQ        MO=AUG E=AUG
PQ        MO=SEP E=SEP
PQ        MO=OCT E=OCT
PQ        MO=NOV-DEC E=JAN
..
LINK      INFLOW    S_SOURCE  SUX
LD        Inflow to Sioux City
IN        B=SUX F=
..
LINK      CHANNEL   SUX       OMA
LD        Channel from Sioux City to Omaha
PQ        MO=JAN-MAR B=SUX-OMA C=FLOW-PNLTY_EDT E=JAN F=
PQ        MO=APR E=APR
PQ        MO=MAY E=MAY
PQ        MO=JUN E=JUN
PQ        MO=JUL-AUG E=JUL
PQ        MO=SEP E=SEP
PQ        MO=OCT E=OCT
PQ        MO=NOV E=NOV
PQ        MO=DEC E=DEC
..
LINK      INFLOW    S_SOURCE  OMA
LD        Inflow to Omaha
IN        B=OMA F=
..
LINK      CHANNEL   OMA       NCNE
LD        Channel from Omaha to Nebraska City
PQ        MO=JAN-MAR B=OMA-NCNE C=FLOW-PNLTY_EDT E=JAN F=
PQ        MO=APR-NOV E=APR
PQ        MO=DEC E=JAN
..
LINK      INFLOW    S_SOURCE  NCNE
LD        Inflow to Nebraska City
IN        B=NCNE F=
..
LINK      CHANNEL   NCNE      MKC
LD        Channel from Nebraska City to Kansas City
PQ        MO=JAN-MAR B=NCNE-MKC C=FLOW-PNLTY_EDT E=JAN F=
PQ        MO=APR-JUN E=APR
PQ        MO=JUL-SEP E=JUL
PQ        MO=OCT-NOV E=OCT
PQ        MO=DEC E=DEC
..
LINK      INFLOW    S_SOURCE  MKC
LD        Inflow to Kansas City
IN        B=MKC F=
..
LINK      CHANNEL   MKC       BNMO
LD        Channel from Kansas City to St. Louis
PQ        MO=JAN-FEB B=MKC-BNMO C=FLOW-PNLTY_EDT E=JAN F=
PQ        MO=MAR E=MAR
PQ        MO=APR E=APR
PQ        MO=MAY E=MAY
PQ        MO=JUN E=JUN
PQ        MO=JUL-AUG E=JUL
PQ        MO=SEP E=SEP
PQ        MO=OCT E=OCT
PQ        MO=NOV E=NOV
PQ        MO=DEC E=DEC
..
LINK      INFLOW    S_SOURCE  BNMO
LD        Inflow to Boonville
IN        B=BNMO F=
..
```

```
LINK      CHANNEL    BNMO      HEMO
LD        Channel from Boonville to Hermann
PQ        MO=JAN-MAR B=BNMO-HEMO C=FLOW-PNLTY_EDT E=JAN F=
PQ        MO=APR E=APR
PQ        MO=MAY E=MAY
PQ        MO=JUN E=JUN
PQ        MO=JUL-SEP E=JUL
PQ        MO=OCT E=OCT
PQ        MO=NOV-DEC E=NOV
..
LINK      INFLOW     S_SOURCE  HEMO
LD        Inflow to Hermann
IN        B=HEMO F=
..
LINK      CHANNEL    HEMO      STL
LD        Channel from Hermann to St. Louis (Sink for now)
PQ        MO=JAN-MAR B=HEMO-STL C=FLOW-PNLTY_EDT E=JAN F=
PQ        MO=APR E=APR
PQ        MO=MAY E=MAY
PQ        MO=JUN E=JUN
PQ        MO=JUL-SEP E=JUL
PQ        MO=OCT E=OCT
PQ        MO=NOV-DEC E=NOV
..
LINK      CHANNEL    STL       S_SINK
LD        Channel from St. Louis to the Super Sink
PQ        MO=JAN-DEC B=STL-S_SINK C=FLOW-PNLTY_EDT E=JAN F=
..
STOP
```

{Print Network Definition Data}

21 Nodes in this system

#	ID	Description
1	S_SOURCE	Super Source
2	S_SINK	Super Sink
3	FTPK	Fort Peck
4	FTPKR	Fort Peck
5	GARR	Garrison
6	GARRR	Garrison
7	OAHE	Oahe
8	OAHER	Oahe
9	BEND	Big Bend
10	BENDR	Big Bend
11	FTRA	Fort Randall
12	FTRAR	Fort Randall
13	GAPT	Gavins Point
14	GAPTR	Gavins Point
15	SUX	Sioux City
16	OMA	Omaha
17	NCNE	Nebraska City
18	MKC	Kansas City
19	BNMO	Boonville
20	HEMO	Hermann
21	STL	St. Louis

6 Reservoirs in this system

#	ID	Link ID	STO1	STOK
1	FTPK	9	13900.00	0.0110000
2	GARR	13	15605.00	0.0130000
3	OAHE	17	17739.00	0.0120000
4	BEND	20	1730.00	0.0240000
5	FTRA	24	3473.00	0.0130000
6	GAPT	28	357.00	0.0330000

43 Links in this system

#	From	To	Type	Mult	Cost	Sum Neg	Num Neg	Description
1	S_SOURCE	S_SINK	DIVR	1.00000	0.00	0.0	0	Link to balance flow throughout computations
2	FTPK	FTPK	STO1	1.00000	0.00	0.0	0	
3	GARR	GARR	STO1	1.00000	0.00	0.0	0	
4	OAHE	OAHE	STO1	1.00000	0.00	0.0	0	
5	BEND	BEND	STO1	1.00000	0.00	0.0	0	
6	FTRA	FTRA	STO1	1.00000	0.00	0.0	0	
7	GAPT	GAPT	STO1	1.00000	0.00	0.0	0	
8	S_SOURCE	FTPK	INFL	1.00000	0.00	0.0	0	Inflow into Fort Peck Reservoir
9	FTPK	FTPK	RSTO	1.00000	0.00	0.0	0	Storage in Fort Peck Reservoir
10	FTPK	FTPKR	HREL	1.00000	0.00	0.0	0	Hydropower release from Ft. Peck
11	FTPKR	GARR	RREL	1.00000	0.00	0.0	0	Release from Fort Peck Reservoir to Garrison
12	S_SOURCE	GARR	INFL	1.00000	0.00	0.0	0	Inflow into Garrison Reservoir
13	GARR	GARR	RSTO	1.00000	0.00	0.0	0	Storage in Garrison Reservoir
14	GARR	GARRR	HREL	1.00000	0.00	0.0	0	Release from Garrison Reservoir to Oahe (energy)
15	GARRR	OAHE	RREL	1.00000	0.00	0.0	0	Release from Garrison Reservoir to Oahe
16	S_SOURCE	OAHE	INFL	1.00000	0.00	0.0	0	Inflow into Oahe Reservoir
17	OAHE	OAHE	RSTO	1.00000	0.00	0.0	0	Storage in Oahe Reservoir
18	OAHE	OAHER	HREL	1.00000	0.00	0.0	0	Release from Oahe Reservoir to Big Bend Reservoir (energy)
19	OAHER	BEND	RREL	1.00000	0.00	0.0	0	Release from Oahe Reservoir to Big Bend Reservoir
20	BEND	BEND	RSTO	1.00000	0.00	0.0	0	Storage in Big Bend Reservoir
21	BEND	BENDR	HREL	1.00000	0.00	0.0	0	Release from Big Bend Reservoir to Fort Randall Reservoir (energy)
22	BENDR	FTRA	RREL	1.00000	0.00	0.0	0	Release from Big Bend Reservoir to Fort Randall Reservoir
23	S_SOURCE	FTRA	INFL	1.00000	0.00	0.0	0	Inflow into Fort Randall Reservoir
24	FTRA	FTRA	RSTO	1.00000	0.00	0.0	0	Storage in Fort Randall Reservoir
25	FTRA	FTRAR	HREL	1.00000	0.00	0.0	0	Release from Fort Randal Reservoir to Gavins Point Reservoir (energy)
26	FTRAR	GAPT	RREL	1.00000	0.00	0.0	0	Release from Fort Randal Reservoir to Gavins Point Reservoir
27	S_SOURCE	GAPT	INFL	1.00000	0.00	0.0	0	Inflow into Gavins Point Reservoir
28	GAPT	GAPT	RSTO	1.00000	0.00	0.0	0	Storage in Gavins Point Reservoir
29	GAPT	GAPTR	HREL	1.00000	0.00	0.0	0	Release from Gavins Point to Sioux City (energy)
30	GAPTR	SUX	RREL	1.00000	0.00	0.0	0	Release from Gavins Point to Sioux City
31	S_SOURCE	SUX	INFL	1.00000	0.00	0.0	0	Inflow to Sioux City
32	SUX	OMA	CHAN	1.00000	0.00	0.0	0	Channel from Sioux City to Omaha
33	S_SOURCE	OMA	INFL	1.00000	0.00	0.0	0	Inflow to Omaha
34	OMA	NCNE	CHAN	1.00000	0.00	0.0	0	Channel from Omaha to Nebraska City
35	S_SOURCE	NCNE	INFL	1.00000	0.00	0.0	0	Inflow to Nebraska City
36	NCNE	MKC	CHAN	1.00000	0.00	0.0	0	Channel from Nebraska City to Kansas City
37	S_SOURCE	MKC	INFL	1.00000	0.00	0.0	0	Inflow to Kansas City
38	MKC	BNMO	CHAN	1.00000	0.00	0.0	0	Channel from Kansas City to St. Louis
39	S_SOURCE	BNMO	INFL	1.00000	0.00	0.0	0	Inflow to Boonville
40	BNMO	HEMO	CHAN	1.00000	0.00	0.0	0	Channel from Boonville to Hermann
41	S_SOURCE	HEMO	INFL	1.00000	0.00	0.0	0	Inflow to Hermann
42	HEMO	STL	CHAN	1.00000	0.00	0.0	0	Channel from Hermann to St. Louis (Sink for now)
43	STL	S_SINK	CHAN	1.00000	0.00	0.0	0	Channel from St. Louis to the Super Sink

167 Pathnames in User Input

#	Link #	Type	Strt Mo	End Mo	Pathname
1	8	IN			//FTPK/FLOW_LOC//1MON//
2	9	PS	1	3	//FTPK/STOR-PNLTY_EDT//JAN//
3	9	PS	4	6	//FTPK/STOR-PNLTY_EDT//APR//
4	9	PS	7	7	//FTPK/STOR-PNLTY_EDT//JUL//
5	9	PS	8	8	//FTPK/STOR-PNLTY_EDT//APR//
6	9	PS	9	9	//FTPK/STOR-PNLTY_EDT//SEP//
7	9	PS	10	10	//FTPK/STOR-PNLTY_EDT//APR//
8	9	PS	11	11	//FTPK/STOR-PNLTY_EDT//SEP//
9	9	PS	12	12	//FTPK/STOR-PNLTY_EDT//DEC//
10	9	EV			//FTPK/EVAP_RATE/71MON//
11	10	PQ	1	12	//FTPK-GARR7FLOW-PNLTY_HPE_EDT////
12	11	PQ	1	3	//FTPK-GARR/FLOW-PNLTY_EDT7/JAN//
13	11	PQ	4	4	//FTPK-GARR/FLOW-PNLTY_EDT//APR//
14	11	PQ	5	6	//FTPK-GARR/FLOW-PNLTY_EDT//MAY//
15	11	PQ	7	7	//FTPK-GARR/FLOW-PNLTY_EDT//JUL//
16	11	PQ	8	8	//FTPK-GARR/FLOW-PNLTY_EDT//AUG//
17	11	PQ	9	9	//FTPK-GARR/FLOW-PNLTY_EDT//SEP//
18	11	PQ	10	10	//FTPK-GARR/FLOW-PNLTY_EDT//OCT//
19	11	PQ	11	12	//FTPK-GARR/FLOW-PNLTY_EDT//JAN//
20	12	IN			//GARR/FLOW_LOC//1MON/7
21	13	PS	1	3	//GARR/STOR-PNLTY_EDT//JAN//
22	13	PS	4	4	//GARR/STOR-PNLTY_EDT//APR//
23	13	PS	5	5	//GARR/STOR-PNLTY_EDT//MAY//
24	13	PS	6	6	//GARR/STOR-PNLTY_EDT//JUN//
25	13	PS	7	7	//GARR/STOR-PNLTY_EDT//JUL//
26	13	PS	8	8	//GARR/STOR-PNLTY_EDT//AUG//
27	13	PS	9	9	//GARR/STOR-PNLTY_EDT//SEP//
28	13	PS	10	10	//GARR/STOR-PNLTY_EDT//OCT//
29	13	PS	11	11	//GARR/STOR-PNLTY_EDT//NOV//
30	13	PS	12	12	//GARR/STOR-PNLTY_EDT//JAN//
31	13	EV			//GARR/EVAP_RATE/71MON//
32	14	PQ	1	12	//GARR-OAHE7FLOW-PNLTY_HPE_EDT////
33	15	PQ	1	3	//GARR-OAHE/FLOW-PNLTY_EDT7/JAN//
34	15	PQ	4	4	//GARR-OAHE/FLOW-PNLTY_EDT//APR//
35	15	PQ	5	6	//GARR-OAHE/FLOW-PNLTY_EDT//MAY//
36	15	PQ	7	7	//GARR-OAHE/FLOW-PNLTY_EDT//JUL//
37	15	PQ	8	8	//GARR-OAHE/FLOW-PNLTY_EDT//AUG//
38	15	PQ	9	9	//GARR-OAHE/FLOW-PNLTY_EDT//SEP//
39	15	PQ	10	10	//GARR-OAHE/FLOW-PNLTY_EDT//OCT//
40	15	PQ	11	12	//GARR-OAHE/FLOW-PNLTY_EDT//JAN//
41	16	IN			//OAHE/FLOW_LOC//1MON/7
42	17	PS	1	3	//OAHE/STOR-PNLTY_EDT//JAN//
43	17	PS	4	4	//OAHE/STOR-PNLTY_EDT//APR//
44	17	PS	5	5	//OAHE/STOR-PNLTY_EDT//MAY//
45	17	PS	6	8	//OAHE/STOR-PNLTY_EDT//JUN//
46	17	PS	9	9	//OAHE/STOR-PNLTY_EDT//SEP//
47	17	PS	10	10	//OAHE/STOR-PNLTY_EDT//OCT//
48	17	PS	11	12	//OAHE/STOR-PNLTY_EDT//JAN//
49	17	EV			//OAHE/EVAP_RATE/71MON//
50	18	PQ	1	12	//OAHE-BEND7FLOW-PNLTY_HPE_EDT////
51	19	PQ	1	3	//OAHE-BEND/FLOW-PNLTY_EDT7/JAN//
52	19	PQ	4	4	//OAHE-BEND/FLOW-PNLTY_EDT//APR//
53	19	PQ	5	6	//OAHE-BEND/FLOW-PNLTY_EDT//MAY//
54	19	PQ	7	7	//OAHE-BEND/FLOW-PNLTY_EDT//JUL//
55	19	PQ	8	8	//OAHE-BEND/FLOW-PNLTY_EDT//AUG//
56	19	PQ	9	9	//OAHE-BEND/FLOW-PNLTY_EDT//SEP//
57	19	PQ	10	10	//OAHE-BEND/FLOW-PNLTY_EDT//OCT//
58	19	PQ	11	12	//OAHE-BEND/FLOW-PNLTY_EDT//JAN//
59	20	PS	1	2	//BEND/STOR-PNLTY_EDT/7JAN//
60	20	PS	3	3	//BEND/STOR-PNLTY_EDT//MAR//
61	20	PS	4	4	//BEND/STOR-PNLTY_EDT//APR//
62	20	PS	5	5	//BEND/STOR-PNLTY_EDT//MAY//
63	20	PS	6	6	//BEND/STOR-PNLTY_EDT//APR//
64	20	PS	7	7	//BEND/STOR-PNLTY_EDT//JUL//
65	20	PS	8	8	//BEND/STOR-PNLTY_EDT//JUL//
66	20	PS	9	9	//BEND/STOR-PNLTY_EDT//SEP//
67	20	PS	10	10	//BEND/STOR-PNLTY_EDT//OCT//
68	20	PS	11	11	//BEND/STOR-PNLTY_EDT//MAR//
69	20	PS	12	12	//BEND/STOR-PNLTY_EDT//DEC//
70	20	EV			//OAHE/EVAP_RATE/71MON//
71	21	PQ	1	12	//BEND-FTRA7FLOW-PNLTY_HPE_EDT////
72	22	PQ	1	3	//BEND-FTRA/FLOW-PNLTY_EDT7/JAN//
73	22	PQ	4	4	//BEND-FTRA/FLOW-PNLTY_EDT//APR//
74	22	PQ	5	6	//BEND-FTRA/FLOW-PNLTY_EDT//MAY//
75	22	PQ	7	7	//BEND-FTRA/FLOW-PNLTY_EDT//JUL//
76	22	PQ	8	8	//BEND-FTRA/FLOW-PNLTY_EDT//AUG//
77	22	PQ	9	9	//BEND-FTRA/FLOW-PNLTY_EDT//SEP//
78	22	PQ	10	10	//BEND-FTRA/FLOW-PNLTY_EDT//OCT//
79	22	PQ	11	12	//BEND-FTRA/FLOW-PNLTY_EDT//JAN//
80	23	IN			//FTRA/FLOW_LOC//1MON/7
81	24	PS	1	3	//FTRA/STOR-PNLTY_EDT//JAN//
82	24	PS	4	4	//FTRA/STOR-PNLTY_EDT//APR//
83	24	PS	5	5	//FTRA/STOR-PNLTY_EDT//MAY//
84	24	PS	6	6	//FTRA/STOR-PNLTY_EDT//JUN//
85	24	PS	7	7	//FTRA/STOR-PNLTY_EDT//JUN//
86	24	PS	8	8	//FTRA/STOR-PNLTY_EDT//AUG//
87	24	PS	9	9	//FTRA/STOR-PNLTY_EDT//SEP//
88	24	PS	10	10	//FTRA/STOR-PNLTY_EDT//MAY//

```
 89   24  PS   11   11 //FTRA/STOR-PNLTY_EDT//NOV//
 90   24  PS   12   12 //FTRA/STOR-PNLTY_EDT//JAN//
 91   24  EV           //FTRA/EVAP_RATE//1MON//
 92   25  PQ    1   12 //FTRA-GAPT/FLOW-PNLTY_HPE_EDT////
 93   26  PQ    1    3 //FTRA-GAPT/FLOW-PNLTY_EDT//JAN//
 94   26  PQ    4    4 //FTRA-GAPT/FLOW-PNLTY_EDT//APR//
 95   26  PQ    5    6 //FTRA-GAPT/FLOW-PNLTY_EDT//JUN//
 96   26  PQ    7    7 //FTRA-GAPT/FLOW-PNLTY_EDT//JUL//
 97   26  PQ    8    8 //FTRA-GAPT/FLOW-PNLTY_EDT//AUG//
 98   26  PQ    9    9 //FTRA-GAPT/FLOW-PNLTY_EDT//SEP//
 99   26  PQ   10   10 //FTRA-GAPT/FLOW-PNLTY_EDT//OCT//
100   26  PQ   11   12 //FTRA-GAPT/FLOW-PNLTY_EDT//JAN//
101   27  IN           //GAPT/FLOW_LOC//1MON//
102   28  PS    1    2 //GAPT/STOR-PNLTY_EDT//JAN//
103   28  PS    3    3 //GAPT/STOR-PNLTY_EDT//MAR//
104   28  PS    4    4 //GAPT/STOR-PNLTY_EDT//APR//
105   28  PS    5    6 //GAPT/STOR-PNLTY_EDT//MAY//
106   28  PS    7    7 //GAPT/STOR-PNLTY_EDT//JUL//
107   28  PS    8    8 //GAPT/STOR-PNLTY_EDT//AUG//
108   28  PS    9    9 //GAPT/STOR-PNLTY_EDT//MAR//
109   28  PS   10   12 //GAPT/STOR-PNLTY_EDT//JAN//
110   28  EV           //GAPT/EVAP_RATE//1MON//
111   29  PQ    1   12 //GAPT-SUX/FLOW-PNLTY_HPE_EDT////
112   30  PQ    1    3 //GAPT-SUX/FLOW-PNLTY_EDT//JAN//
113   30  PQ    4    4 //GAPT-SUX/FLOW-PNLTY_EDT//APR//
114   30  PQ    5    6 //GAPT-SUX/FLOW-PNLTY_EDT//MAY//
115   30  PQ    7    7 //GAPT-SUX/FLOW-PNLTY_EDT//JUL//
116   30  PQ    8    8 //GAPT-SUX/FLOW-PNLTY_EDT//AUG//
117   30  PQ    9    9 //GAPT-SUX/FLOW-PNLTY_EDT//SEP//
118   30  PQ   10   10 //GAPT-SUX/FLOW-PNLTY_EDT//OCT//
119   30  PQ   11   12 //GAPT-SUX/FLOW-PNLTY_EDT//JAN//
120   31  IN           //SUX/FLOW_LOC//1MON//
121   32  PQ    1    3 //SUX-OMA/FLOW-PNLTY_EDT//JAN//
122   32  PQ    4    4 //SUX-OMA/FLOW-PNLTY_EDT//APR//
123   32  PQ    5    5 //SUX-OMA/FLOW-PNLTY_EDT//MAY//
124   32  PQ    6    6 //SUX-OMA/FLOW-PNLTY_EDT//JUN//
125   32  PQ    7    8 //SUX-OMA/FLOW-PNLTY_EDT//JUL//
126   32  PQ    9    9 //SUX-OMA/FLOW-PNLTY_EDT//SEP//
127   32  PQ   10   10 //SUX-OMA/FLOW-PNLTY_EDT//OCT//
128   32  PQ   11   11 //SUX-OMA/FLOW-PNLTY_EDT//NOV//
129   32  PQ   12   12 //SUX-OMA/FLOW-PNLTY_EDT//DEC//
130   33  IN           //OMA/FLOW_LOC//1MON//
131   34  PQ    1    3 //OMA-NCNE/FLOW-PNLTY_EDT//JAN//
132   34  PQ    4   11 //OMA-NCNE/FLOW-PNLTY_EDT//APR//
133   34  PQ   12   12 //OMA-NCNE/FLOW-PNLTY_EDT//JAN//
134   35  IN           //NCNE/FLOW_LOC//1MON//
135   36  PQ    1    3 //NCNE-MKC/FLOW-PNLTY_EDT//JAN//
136   36  PQ    4    6 //NCNE-MKC/FLOW-PNLTY_EDT//APR//
137   36  PQ    7    9 //NCNE-MKC/FLOW-PNLTY_EDT//JUL//
138   36  PQ   10   11 //NCNE-MKC/FLOW-PNLTY_EDT//OCT//
139   36  PQ   12   12 //NCNE-MKC/FLOW-PNLTY_EDT//DEC//
140   37  IN           //MKC/FLOW_LOC//1MON//
141   38  PQ    1    2 //MKC-BNMO/FLOW-PNLTY_EDT//JAN//
142   38  PQ    3    3 //MKC-BNMO/FLOW-PNLTY_EDT//MAR//
143   38  PQ    4    4 //MKC-BNMO/FLOW-PNLTY_EDT//APR//
144   38  PQ    5    5 //MKC-BNMO/FLOW-PNLTY_EDT//MAY//
145   38  PQ    6    6 //MKC-BNMO/FLOW-PNLTY_EDT//JUN//
146   38  PQ    7    8 //MKC-BNMO/FLOW-PNLTY_EDT//JUL//
147   38  PQ    9    9 //MKC-BNMO/FLOW-PNLTY_EDT//SEP//
148   38  PQ   10   10 //MKC-BNMO/FLOW-PNLTY_EDT//OCT//
149   38  PQ   11   11 //MKC-BNMO/FLOW-PNLTY_EDT//NOV//
150   38  PQ   12   12 //MKC-BNMO/FLOW-PNLTY_EDT//DEC//
151   39  IN           //BNMO/FLOW_LOC//1MON//
152   40  PQ    1    3 //BNMO-HEMO/FLOW-PNLTY_EDT//JAN//
153   40  PQ    4    4 //BNMO-HEMO/FLOW-PNLTY_EDT//APR//
154   40  PQ    5    5 //BNMO-HEMO/FLOW-PNLTY_EDT//MAY//
155   40  PQ    6    6 //BNMO-HEMO/FLOW-PNLTY_EDT//JUN//
156   40  PQ    7    9 //BNMO-HEMO/FLOW-PNLTY_EDT//JUL//
157   40  PQ   10   10 //BNMO-HEMO/FLOW-PNLTY_EDT//OCT//
158   40  PQ   11   12 //BNMO-HEMO/FLOW-PNLTY_EDT//NOV//
159   41  IN           //HEMO/FLOW_LOC//1MON//
160   42  PQ    1    3 //HEMO-STL/FLOW-PNLTY_EDT//JAN//
161   42  PQ    4    4 //HEMO-STL/FLOW-PNLTY_EDT//APR//
162   42  PQ    5    5 //HEMO-STL/FLOW-PNLTY_EDT//MAY//
163   42  PQ    6    6 //HEMO-STL/FLOW-PNLTY_EDT//JUN//
164   42  PQ    7    9 //HEMO-STL/FLOW-PNLTY_EDT//JUL//
165   42  PQ   10   10 //HEMO-STL/FLOW-PNLTY_EDT//OCT//
166   42  PQ   11   12 //HEMO-STL/FLOW-PNLTY_EDT//NOV//
167   43  PQ    1   12 //STL-S_SINK/FLOW-PNLTY_EDT//JAN//
```

{Read penalty functions and time series data and generate network.}

```
-----DSS--- ZREAD Unit  71; Vers.   1:  //FTPK/FLOW_LOC/01JAN1960/1MON//
-----DSS--- ZREAD Unit  72; Vers.   1:  //FTPK/STOR-PNLTY_EDT//JAN//
-----DSS---ZWRITE Unit  73; Vers.   3:  //FTPK/STOR-PNLTY_EDT//JAN/EXMPL1/
-----DSS---ZWRITE Unit  73; Vers.   3:  //FTPK/STOR-PNLTY_EDT//FEB/EXMPL1/
```

```
-----DSS---ZWRITE Unit  73; Vers.   3:  //FTPK/STOR-PNLTY_EDT//MAR/EXMPL1/
-----DSS--- ZREAD Unit  72; Vers.   1:  //FTPK/STOR-PNLTY_EDT//APR//
-----DSS---ZWRITE Unit  73; Vers.   3:  //FTPK/STOR-PNLTY_EDT//APR/EXMPL1/
-----DSS---ZWRITE Unit  73; Vers.   3:  //FTPK/STOR-PNLTY_EDT//MAY/EXMPL1/
-----DSS---ZWRITE Unit  73; Vers.   3:  //FTPK/STOR-PNLTY_EDT//JUN/EXMPL1/
-----DSS--- ZREAD Unit  72; Vers.   1:  //FTPK/STOR-PNLTY_EDT//JUL//
-----DSS---ZWRITE Unit  73; Vers.   3:  //FTPK/STOR-PNLTY_EDT//JUL/EXMPL1/
-----DSS--- ZREAD Unit  72; Vers.   1:  //FTPK/STOR-PNLTY_EDT//APR//
-----DSS---ZWRITE Unit  73; Vers.   3:  //FTPK/STOR-PNLTY_EDT//AUG/EXMPL1/
-----DSS--- ZREAD Unit  72; Vers.   1:  //FTPK/STOR-PNLTY_EDT//SEP//
-----DSS---ZWRITE Unit  73; Vers.   3:  //FTPK/STOR-PNLTY_EDT//SEP/EXMPL1/
-----DSS--- ZREAD Unit  72; Vers.   1:  //FTPK/STOR-PNLTY_EDT//APR//
-----DSS---ZWRITE Unit  73; Vers.   3:  //FTPK/STOR-PNLTY_EDT//OCT/EXMPL1/
-----DSS--- ZREAD Unit  72; Vers.   1:  //FTPK/STOR-PNLTY_EDT//SEP//
-----DSS---ZWRITE Unit  73; Vers.   3:  //FTPK/STOR-PNLTY_EDT//NOV/EXMPL1/
-----DSS--- ZREAD Unit  72; Vers.   1:  //FTPK/STOR-PNLTY_EDT//DEC//
-----DSS---ZWRITE Unit  73; Vers.   3:  //FTPK/STOR-PNLTY_EDT//DEC/EXMPL1/
-----DSS--- ZREAD Unit  71; Vers.   2:  //FTPK/EVAP_RATE/01JAN1960/1MON//
-----DSS--- ZREAD Unit  72; Vers.   2:  //FTPK-GARR/FLOW-PNLTY_HPE_EDT////
-----DSS---ZWRITE Unit  73; Vers.   3:  //FTPK-FTPKR/FLOW-PNLTY_HPE_EDT//JAN/EXMPL1/
-----DSS---ZWRITE Unit  73; Vers.   3:  //FTPK-FTPKR/FLOW-PNLTY_HPE_EDT//FEB/EXMPL1/
-----DSS---ZWRITE Unit  73; Vers.   3:  //FTPK-FTPKR/FLOW-PNLTY_HPE_EDT//MAR/EXMPL1/
-----DSS---ZWRITE Unit  73; Vers.   3:  //FTPK-FTPKR/FLOW-PNLTY_HPE_EDT//APR/EXMPL1/
-----DSS---ZWRITE Unit  73; Vers.   3:  //FTPK-FTPKR/FLOW-PNLTY_HPE_EDT//MAY/EXMPL1/
-----DSS---ZWRITE Unit  73; Vers.   3:  //FTPK-FTPKR/FLOW-PNLTY_HPE_EDT//JUN/EXMPL1/
-----DSS---ZWRITE Unit  73; Vers.   3:  //FTPK-FTPKR/FLOW-PNLTY_HPE_EDT//JUL/EXMPL1/
-----DSS---ZWRITE Unit  73; Vers.   3:  //FTPK-FTPKR/FLOW-PNLTY_HPE_EDT//AUG/EXMPL1/
-----DSS---ZWRITE Unit  73; Vers.   3:  //FTPK-FTPKR/FLOW-PNLTY_HPE_EDT//SEP/EXMPL1/
-----DSS---ZWRITE Unit  73; Vers.   3:  //FTPK-FTPKR/FLOW-PNLTY_HPE_EDT//OCT/EXMPL1/
-----DSS---ZWRITE Unit  73; Vers.   3:  //FTPK-FTPKR/FLOW-PNLTY_HPE_EDT//NOV/EXMPL1/
-----DSS---ZWRITE Unit  73; Vers.   3:  //FTPK-FTPKR/FLOW-PNLTY_HPE_EDT//DEC/EXMPL1/
-----DSS--- ZREAD Unit  72; Vers.   3:  //FTPK-GARR/FLOW-PNLTY_EDT//JAN//
-----DSS---ZWRITE Unit  73; Vers.   3:  //FTPKR-GARR/FLOW-PNLTY_EDT//JAN/EXMPL1/
-----DSS---ZWRITE Unit  73; Vers.   3:  //FTPKR-GARR/FLOW-PNLTY_EDT//FEB/EXMPL1/
-----DSS---ZWRITE Unit  73; Vers.   3:  //FTPKR-GARR/FLOW-PNLTY_EDT//MAR/EXMPL1/
-----DSS--- ZREAD Unit  72; Vers.   2:  //FTPK-GARR/FLOW-PNLTY_EDT//APR//
-----DSS---ZWRITE Unit  73; Vers.   3:  //FTPKR-GARR/FLOW-PNLTY_EDT//APR/EXMPL1/
-----DSS--- ZREAD Unit  72; Vers.   2:  //FTPK-GARR/FLOW-PNLTY_EDT//MAY//
-----DSS---ZWRITE Unit  73; Vers.   3:  //FTPKR-GARR/FLOW-PNLTY_EDT//MAY/EXMPL1/
-----DSS---ZWRITE Unit  73; Vers.   3:  //FTPKR-GARR/FLOW-PNLTY_EDT//JUN/EXMPL1/
-----DSS--- ZREAD Unit  72; Vers.   2:  //FTPK-GARR/FLOW-PNLTY_EDT//JUL//
-----DSS---ZWRITE Unit  73; Vers.   3:  //FTPKR-GARR/FLOW-PNLTY_EDT//JUL/EXMPL1/
-----DSS--- ZREAD Unit  72; Vers.   2:  //FTPK-GARR/FLOW-PNLTY_EDT//AUG//
-----DSS---ZWRITE Unit  73; Vers.   3:  //FTPKR-GARR/FLOW-PNLTY_EDT//AUG/EXMPL1/
-----DSS--- ZREAD Unit  72; Vers.   2:  //FTPK-GARR/FLOW-PNLTY_EDT//SEP//
-----DSS---ZWRITE Unit  73; Vers.   3:  //FTPKR-GARR/FLOW-PNLTY_EDT//SEP/EXMPL1/
-----DSS--- ZREAD Unit  72; Vers.   2:  //FTPK-GARR/FLOW-PNLTY_EDT//OCT//
-----DSS---ZWRITE Unit  73; Vers.   3:  //FTPKR-GARR/FLOW-PNLTY_EDT//OCT/EXMPL1/
-----DSS--- ZREAD Unit  72; Vers.   3:  //FTPK-GARR/FLOW-PNLTY_EDT//JAN//
-----DSS---ZWRITE Unit  73; Vers.   3:  //FTPKR-GARR/FLOW-PNLTY_EDT//NOV/EXMPL1/
-----DSS---ZWRITE Unit  73; Vers.   3:  //FTPKR-GARR/FLOW-PNLTY_EDT//DEC/EXMPL1/
-----DSS--- ZREAD Unit  71; Vers.   1:  //GARR/FLOW_LOC/01JAN1960/1MON//
-----DSS--- ZREAD Unit  72; Vers.   1:  //GARR/STOR-PNLTY_EDT//JAN//
-----DSS---ZWRITE Unit  73; Vers.   3:  //GARR/STOR-PNLTY_EDT//JAN/EXMPL1/
-----DSS---ZWRITE Unit  73; Vers.   3:  //GARR/STOR-PNLTY_EDT//FEB/EXMPL1/
-----DSS---ZWRITE Unit  73; Vers.   3:  //GARR/STOR-PNLTY_EDT//MAR/EXMPL1/
-----DSS--- ZREAD Unit  72; Vers.   1:  //GARR/STOR-PNLTY_EDT//APR//
-----DSS---ZWRITE Unit  73; Vers.   3:  //GARR/STOR-PNLTY_EDT//APR/EXMPL1/
-----DSS--- ZREAD Unit  72; Vers.   1:  //GARR/STOR-PNLTY_EDT//MAY//
-----DSS---ZWRITE Unit  73; Vers.   3:  //GARR/STOR-PNLTY_EDT//MAY/EXMPL1/
-----DSS--- ZREAD Unit  72; Vers.   1:  //GARR/STOR-PNLTY_EDT//JUN//
-----DSS---ZWRITE Unit  73; Vers.   3:  //GARR/STOR-PNLTY_EDT//JUN/EXMPL1/
-----DSS--- ZREAD Unit  72; Vers.   1:  //GARR/STOR-PNLTY_EDT//JUL//
-----DSS---ZWRITE Unit  73; Vers.   3:  //GARR/STOR-PNLTY_EDT//JUL/EXMPL1/
-----DSS--- ZREAD Unit  72; Vers.   1:  //GARR/STOR-PNLTY_EDT//AUG//
-----DSS---ZWRITE Unit  73; Vers.   3:  //GARR/STOR-PNLTY_EDT//AUG/EXMPL1/
-----DSS--- ZREAD Unit  72; Vers.   1:  //GARR/STOR-PNLTY_EDT//SEP//
-----DSS---ZWRITE Unit  73; Vers.   3:  //GARR/STOR-PNLTY_EDT//SEP/EXMPL1/
-----DSS--- ZREAD Unit  72; Vers.   1:  //GARR/STOR-PNLTY_EDT//OCT//
-----DSS---ZWRITE Unit  73; Vers.   3:  //GARR/STOR-PNLTY_EDT//OCT/EXMPL1/
-----DSS--- ZREAD Unit  72; Vers.   1:  //GARR/STOR-PNLTY_EDT//NOV//
-----DSS---ZWRITE Unit  73; Vers.   3:  //GARR/STOR-PNLTY_EDT//NOV/EXMPL1/
-----DSS--- ZREAD Unit  72; Vers.   1:  //GARR/STOR-PNLTY_EDT//JAN//
-----DSS---ZWRITE Unit  73; Vers.   3:  //GARR/STOR-PNLTY_EDT//DEC/EXMPL1/
-----DSS--- ZREAD Unit  71; Vers.   2:  //GARR/EVAP_RATE/01JAN1960/1MON//
-----DSS--- ZREAD Unit  72; Vers.   2:  //GARR-OAHE/FLOW-PNLTY_HPE_EDT////
-----DSS---ZWRITE Unit  73; Vers.   3:  //GARR-GARRR/FLOW-PNLTY_HPE_EDT//JAN/EXMPL1/
-----DSS---ZWRITE Unit  73; Vers.   3:  //GARR-GARRR/FLOW-PNLTY_HPE_EDT//FEB/EXMPL1/
-----DSS---ZWRITE Unit  73; Vers.   3:  //GARR-GARRR/FLOW-PNLTY_HPE_EDT//MAR/EXMPL1/
-----DSS---ZWRITE Unit  73; Vers.   3:  //GARR-GARRR/FLOW-PNLTY_HPE_EDT//APR/EXMPL1/
-----DSS---ZWRITE Unit  73; Vers.   3:  //GARR-GARRR/FLOW-PNLTY_HPE_EDT//MAY/EXMPL1/
-----DSS---ZWRITE Unit  73; Vers.   3:  //GARR-GARRR/FLOW-PNLTY_HPE_EDT//JUN/EXMPL1/
-----DSS---ZWRITE Unit  73; Vers.   3:  //GARR-GARRR/FLOW-PNLTY_HPE_EDT//JUL/EXMPL1/
-----DSS---ZWRITE Unit  73; Vers.   3:  //GARR-GARRR/FLOW-PNLTY_HPE_EDT//AUG/EXMPL1/
-----DSS---ZWRITE Unit  73; Vers.   3:  //GARR-GARRR/FLOW-PNLTY_HPE_EDT//SEP/EXMPL1/
-----DSS---ZWRITE Unit  73; Vers.   3:  //GARR-GARRR/FLOW-PNLTY_HPE_EDT//OCT/EXMPL1/
-----DSS---ZWRITE Unit  73; Vers.   3:  //GARR-GARRR/FLOW-PNLTY_HPE_EDT//NOV/EXMPL1/
-----DSS---ZWRITE Unit  73; Vers.   3:  //GARR-GARRR/FLOW-PNLTY_HPE_EDT//DEC/EXMPL1/
-----DSS--- ZREAD Unit  72; Vers.   2:  //GARR-OAHE/FLOW-PNLTY_EDT//JAN//
-----DSS---ZWRITE Unit  73; Vers.   3:  //GARRR-OAHE/FLOW-PNLTY_EDT//JAN/EXMPL1/
-----DSS---ZWRITE Unit  73; Vers.   3:  //GARRR-OAHE/FLOW-PNLTY_EDT//FEB/EXMPL1/
-----DSS---ZWRITE Unit  73; Vers.   3:  //GARRR-OAHE/FLOW-PNLTY_EDT//MAR/EXMPL1/
```

```
-----DSS--- ZREAD Unit  72; Vers.    2:  //GARR-OAHE/FLOW-PNLTY_EDT//APR//
-----DSS---ZWRITE Unit  73; Vers.    3:  //GARRR-OAHE/FLOW-PNLTY_EDT//APR/EXMPL1/
-----DSS--- ZREAD Unit  72; Vers.    2:  //GARR-OAHE/FLOW-PNLTY_EDT//MAY//
-----DSS---ZWRITE Unit  73; Vers.    3:  //GARRR-OAHE/FLOW-PNLTY_EDT//MAY/EXMPL1/
-----DSS---ZWRITE Unit  73; Vers.    3:  //GARRR-OAHE/FLOW-PNLTY_EDT//JUN/EXMPL1/
-----DSS--- ZREAD Unit  72; Vers.    2:  //GARR-OAHE/FLOW-PNLTY_EDT//JUL//
-----DSS---ZWRITE Unit  73; Vers.    3:  //GARRR-OAHE/FLOW-PNLTY_EDT//JUL/EXMPL1/
-----DSS--- ZREAD Unit  72; Vers.    2:  //GARR-OAHE/FLOW-PNLTY_EDT//AUG//
-----DSS---ZWRITE Unit  73; Vers.    3:  //GARRR-OAHE/FLOW-PNLTY_EDT//AUG/EXMPL1/
-----DSS--- ZREAD Unit  72; Vers.    2:  //GARR-OAHE/FLOW-PNLTY_EDT//SEP//
-----DSS---ZWRITE Unit  73; Vers.    3:  //GARRR-OAHE/FLOW-PNLTY_EDT//SEP/EXMPL1/
-----DSS--- ZREAD Unit  72; Vers.    2:  //GARR-OAHE/FLOW-PNLTY_EDT//OCT//
-----DSS---ZWRITE Unit  73; Vers.    3:  //GARRR-OAHE/FLOW-PNLTY_EDT//OCT/EXMPL1/
-----DSS--- ZREAD Unit  72; Vers.    2:  //GARR-OAHE/FLOW-PNLTY_EDT//JAN//
-----DSS---ZWRITE Unit  73; Vers.    3:  //GARRR-OAHE/FLOW-PNLTY_EDT//NOV/EXMPL1/
-----DSS---ZWRITE Unit  73; Vers.    3:  //GARRR-OAHE/FLOW-PNLTY_EDT//DEC/EXMPL1/
-----DSS--- ZREAD Unit  71; Vers.    1:  //OAHE/FLOW_LOC/01JAN1960/1MON//


NUMNEG:         3         SUMNEG:    -266.000         SUMNEG:    -0.266000E+03
Path: //OAHE/FLOW_LOC//1MON//


-----DSS--- ZREAD Unit  72; Vers.    1:  //OAHE/STOR-PNLTY_EDT//JAN//
-----DSS---ZWRITE Unit  73; Vers.    3:  //OAHE/STOR-PNLTY_EDT//JAN/EXMPL1/
-----DSS---ZWRITE Unit  73; Vers.    3:  //OAHE/STOR-PNLTY_EDT//FEB/EXMPL1/
-----DSS---ZWRITE Unit  73; Vers.    3:  //OAHE/STOR-PNLTY_EDT//MAR/EXMPL1/
-----DSS--- ZREAD Unit  72; Vers.    1:  //OAHE/STOR-PNLTY_EDT//APR//
-----DSS---ZWRITE Unit  73; Vers.    3:  //OAHE/STOR-PNLTY_EDT//APR/EXMPL1/
-----DSS--- ZREAD Unit  72; Vers.    1:  //OAHE/STOR-PNLTY_EDT//MAY//
-----DSS---ZWRITE Unit  73; Vers.    3:  //OAHE/STOR-PNLTY_EDT//MAY/EXMPL1/
-----DSS--- ZREAD Unit  72; Vers.    1:  //OAHE/STOR-PNLTY_EDT//JUN//
-----DSS---ZWRITE Unit  73; Vers.    3:  //OAHE/STOR-PNLTY_EDT//JUN/EXMPL1/
-----DSS---ZWRITE Unit  73; Vers.    3:  //OAHE/STOR-PNLTY_EDT//JUL/EXMPL1/
-----DSS---ZWRITE Unit  73; Vers.    3:  //OAHE/STOR-PNLTY_EDT//AUG/EXMPL1/
-----DSS--- ZREAD Unit  72; Vers.    1:  //OAHE/STOR-PNLTY_EDT//SEP//
-----DSS---ZWRITE Unit  73; Vers.    3:  //OAHE/STOR-PNLTY_EDT//SEP/EXMPL1/
-----DSS--- ZREAD Unit  72; Vers.    1:  //OAHE/STOR-PNLTY_EDT//OCT//
-----DSS---ZWRITE Unit  73; Vers.    3:  //OAHE/STOR-PNLTY_EDT//OCT/EXMPL1/
-----DSS--- ZREAD Unit  72; Vers.    1:  //OAHE/STOR-PNLTY_EDT//JAN//
-----DSS---ZWRITE Unit  73; Vers.    3:  //OAHE/STOR-PNLTY_EDT//NOV/EXMPL1/
-----DSS---ZWRITE Unit  73; Vers.    3:  //OAHE/STOR-PNLTY_EDT//DEC/EXMPL1/
-----DSS--- ZREAD Unit  71; Vers.    2:  //OAHE/EVAP_RATE/01JAN1960/1MON//
-----DSS--- ZREAD Unit  72; Vers.    2:  //OAHE-BEND/FLOW-PNLTY_HPE_EDT////
-----DSS---ZWRITE Unit  73; Vers.    3:  //OAHE-OAHER/FLOW-PNLTY_HPE_EDT//JAN/EXMPL1/
-----DSS---ZWRITE Unit  73; Vers.    3:  //OAHE-OAHER/FLOW-PNLTY_HPE_EDT//FEB/EXMPL1/
-----DSS---ZWRITE Unit  73; Vers.    3:  //OAHE-OAHER/FLOW-PNLTY_HPE_EDT//MAR/EXMPL1/
-----DSS---ZWRITE Unit  73; Vers.    3:  //OAHE-OAHER/FLOW-PNLTY_HPE_EDT//APR/EXMPL1/
-----DSS---ZWRITE Unit  73; Vers.    3:  //OAHE-OAHER/FLOW-PNLTY_HPE_EDT//MAY/EXMPL1/
-----DSS---ZWRITE Unit  73; Vers.    3:  //OAHE-OAHER/FLOW-PNLTY_HPE_EDT//JUN/EXMPL1/
-----DSS---ZWRITE Unit  73; Vers.    3:  //OAHE-OAHER/FLOW-PNLTY_HPE_EDT//JUL/EXMPL1/
-----DSS---ZWRITE Unit  73; Vers.    3:  //OAHE-OAHER/FLOW-PNLTY_HPE_EDT//AUG/EXMPL1/
-----DSS---ZWRITE Unit  73; Vers.    3:  //OAHE-OAHER/FLOW-PNLTY_HPE_EDT//SEP/EXMPL1/
-----DSS---ZWRITE Unit  73; Vers.    3:  //OAHE-OAHER/FLOW-PNLTY_HPE_EDT//OCT/EXMPL1/
-----DSS---ZWRITE Unit  73; Vers.    3:  //OAHE-OAHER/FLOW-PNLTY_HPE_EDT//NOV/EXMPL1/
-----DSS---ZWRITE Unit  73; Vers.    3:  //OAHE-OAHER/FLOW-PNLTY_HPE_EDT//DEC/EXMPL1/
-----DSS--- ZREAD Unit  72; Vers.    2:  //OAHE-BEND/FLOW-PNLTY_EDT//JAN//
-----DSS---ZWRITE Unit  73; Vers.    3:  //OAHER-BEND/FLOW-PNLTY_EDT//JAN/EXMPL1/
-----DSS---ZWRITE Unit  73; Vers.    3:  //OAHER-BEND/FLOW-PNLTY_EDT//FEB/EXMPL1/
-----DSS---ZWRITE Unit  73; Vers.    3:  //OAHER-BEND/FLOW-PNLTY_EDT//MAR/EXMPL1/
-----DSS--- ZREAD Unit  72; Vers.    2:  //OAHE-BEND/FLOW-PNLTY_EDT//APR//
-----DSS---ZWRITE Unit  73; Vers.    3:  //OAHER-BEND/FLOW-PNLTY_EDT//APR/EXMPL1/
-----DSS--- ZREAD Unit  72; Vers.    2:  //OAHE-BEND/FLOW-PNLTY_EDT//MAY//
-----DSS---ZWRITE Unit  73; Vers.    3:  //OAHER-BEND/FLOW-PNLTY_EDT//MAY/EXMPL1/
-----DSS---ZWRITE Unit  73; Vers.    3:  //OAHER-BEND/FLOW-PNLTY_EDT//JUN/EXMPL1/
-----DSS--- ZREAD Unit  72; Vers.    2:  //OAHE-BEND/FLOW-PNLTY_EDT//JUL//
-----DSS---ZWRITE Unit  73; Vers.    3:  //OAHER-BEND/FLOW-PNLTY_EDT//JUL/EXMPL1/
-----DSS--- ZREAD Unit  72; Vers.    2:  //OAHE-BEND/FLOW-PNLTY_EDT//AUG//
-----DSS---ZWRITE Unit  73; Vers.    3:  //OAHER-BEND/FLOW-PNLTY_EDT//AUG/EXMPL1/
-----DSS--- ZREAD Unit  72; Vers.    2:  //OAHE-BEND/FLOW-PNLTY_EDT//SEP//
-----DSS---ZWRITE Unit  73; Vers.    3:  //OAHER-BEND/FLOW-PNLTY_EDT//SEP/EXMPL1/
-----DSS--- ZREAD Unit  72; Vers.    2:  //OAHE-BEND/FLOW-PNLTY_EDT//OCT//
-----DSS---ZWRITE Unit  73; Vers.    3:  //OAHER-BEND/FLOW-PNLTY_EDT//OCT/EXMPL1/
-----DSS--- ZREAD Unit  72; Vers.    2:  //OAHE-BEND/FLOW-PNLTY_EDT//JAN//
-----DSS---ZWRITE Unit  73; Vers.    3:  //OAHER-BEND/FLOW-PNLTY_EDT//NOV/EXMPL1/
-----DSS---ZWRITE Unit  73; Vers.    3:  //OAHER-BEND/FLOW-PNLTY_EDT//DEC/EXMPL1/
-----DSS--- ZREAD Unit  72; Vers.    1:  //BEND/STOR-PNLTY_EDT//JAN//
-----DSS---ZWRITE Unit  73; Vers.    3:  //BEND/STOR-PNLTY_EDT//JAN/EXMPL1/
-----DSS---ZWRITE Unit  73; Vers.    3:  //BEND/STOR-PNLTY_EDT//FEB/EXMPL1/
-----DSS--- ZREAD Unit  72; Vers.    1:  //BEND/STOR-PNLTY_EDT//MAR//
-----DSS---ZWRITE Unit  73; Vers.    3:  //BEND/STOR-PNLTY_EDT//MAR/EXMPL1/
-----DSS--- ZREAD Unit  72; Vers.    1:  //BEND/STOR-PNLTY_EDT//APR//
-----DSS---ZWRITE Unit  73; Vers.    3:  //BEND/STOR-PNLTY_EDT//APR/EXMPL1/
-----DSS--- ZREAD Unit  72; Vers.    1:  //BEND/STOR-PNLTY_EDT//MAY//
-----DSS---ZWRITE Unit  73; Vers.    3:  //BEND/STOR-PNLTY_EDT//MAY/EXMPL1/
-----DSS--- ZREAD Unit  72; Vers.    1:  //BEND/STOR-PNLTY_EDT//APR//
-----DSS---ZWRITE Unit  73; Vers.    3:  //BEND/STOR-PNLTY_EDT//JUN/EXMPL1/
-----DSS--- ZREAD Unit  72; Vers.    1:  //BEND/STOR-PNLTY_EDT//JUL//
-----DSS---ZWRITE Unit  73; Vers.    3:  //BEND/STOR-PNLTY_EDT//JUL/EXMPL1/
-----DSS--- ZREAD Unit  72; Vers.    1:  //BEND/STOR-PNLTY_EDT//JUL//
-----DSS---ZWRITE Unit  73; Vers.    3:  //BEND/STOR-PNLTY_EDT//AUG/EXMPL1/
-----DSS--- ZREAD Unit  72; Vers.    1:  //BEND/STOR-PNLTY_EDT//SEP//
```

```
-----DSS---ZWRITE Unit  73; Vers.   3:  //BEND/STOR-PNLTY_EDT//SEP/EXMPL1/
-----DSS--- ZREAD Unit  72; Vers.   1:  //BEND/STOR-PNLTY_EDT//OCT//
-----DSS---ZWRITE Unit  73; Vers.   3:  //BEND/STOR-PNLTY_EDT//OCT/EXMPL1/
-----DSS--- ZREAD Unit  72; Vers.   1:  //BEND/STOR-PNLTY_EDT//MAR//
-----DSS---ZWRITE Unit  73; Vers.   3:  //BEND/STOR-PNLTY_EDT//NOV/EXMPL1/
-----DSS--- ZREAD Unit  72; Vers.   1:  //BEND/STOR-PNLTY_EDT//DEC//
-----DSS---ZWRITE Unit  73; Vers.   3:  //BEND/STOR-PNLTY_EDT//DEC/EXMPL1/
-----DSS--- ZREAD Unit  71; Vers.   2:  //OAHE/EVAP_RATE/01JAN1960/1MON//
-----DSS--- ZREAD Unit  72; Vers.   2:  //BEND-FTRA/FLOW-PNLTY_HPE_EDT////
-----DSS---ZWRITE Unit  73; Vers.   3:  //BEND-BENDR/FLOW-PNLTY_HPE_EDT//JAN/EXMPL1/
-----DSS---ZWRITE Unit  73; Vers.   3:  //BEND-BENDR/FLOW-PNLTY_HPE_EDT//FEB/EXMPL1/
-----DSS---ZWRITE Unit  73; Vers.   3:  //BEND-BENDR/FLOW-PNLTY_HPE_EDT//MAR/EXMPL1/
-----DSS---ZWRITE Unit  73; Vers.   3:  //BEND-BENDR/FLOW-PNLTY_HPE_EDT//APR/EXMPL1/
-----DSS---ZWRITE Unit  73; Vers.   3:  //BEND-BENDR/FLOW-PNLTY_HPE_EDT//MAY/EXMPL1/
-----DSS---ZWRITE Unit  73; Vers.   3:  //BEND-BENDR/FLOW-PNLTY_HPE_EDT//JUN/EXMPL1/
-----DSS---ZWRITE Unit  73; Vers.   3:  //BEND-BENDR/FLOW-PNLTY_HPE_EDT//JUL/EXMPL1/
-----DSS---ZWRITE Unit  73; Vers.   3:  //BEND-BENDR/FLOW-PNLTY_HPE_EDT//AUG/EXMPL1/
-----DSS---ZWRITE Unit  73; Vers.   3:  //BEND-BENDR/FLOW-PNLTY_HPE_EDT//SEP/EXMPL1/
-----DSS---ZWRITE Unit  73; Vers.   3:  //BEND-BENDR/FLOW-PNLTY_HPE_EDT//OCT/EXMPL1/
-----DSS---ZWRITE Unit  73; Vers.   3:  //BEND-BENDR/FLOW-PNLTY_HPE_EDT//NOV/EXMPL1/
-----DSS---ZWRITE Unit  73; Vers.   3:  //BEND-BENDR/FLOW-PNLTY_HPE_EDT//DEC/EXMPL1/
-----DSS--- ZREAD Unit  72; Vers.   2:  //BEND-FTRA/FLOW-PNLTY_EDT//JAN//
-----DSS---ZWRITE Unit  73; Vers.   3:  //BENDR-FTRA/FLOW-PNLTY_EDT//JAN/EXMPL1/
-----DSS---ZWRITE Unit  73; Vers.   3:  //BENDR-FTRA/FLOW-PNLTY_EDT//FEB/EXMPL1/
-----DSS---ZWRITE Unit  73; Vers.   3:  //BENDR-FTRA/FLOW-PNLTY_EDT//MAR/EXMPL1/
-----DSS--- ZREAD Unit  72; Vers.   2:  //BEND-FTRA/FLOW-PNLTY_EDT//APR//
-----DSS---ZWRITE Unit  73; Vers.   3:  //BENDR-FTRA/FLOW-PNLTY_EDT//APR/EXMPL1/
-----DSS--- ZREAD Unit  72; Vers.   2:  //BEND-FTRA/FLOW-PNLTY_EDT//MAY//
-----DSS---ZWRITE Unit  73; Vers.   3:  //BENDR-FTRA/FLOW-PNLTY_EDT//MAY/EXMPL1/
-----DSS---ZWRITE Unit  73; Vers.   3:  //BENDR-FTRA/FLOW-PNLTY_EDT//JUN/EXMPL1/
-----DSS--- ZREAD Unit  72; Vers.   2:  //BEND-FTRA/FLOW-PNLTY_EDT//JUL//
-----DSS---ZWRITE Unit  73; Vers.   3:  //BENDR-FTRA/FLOW-PNLTY_EDT//JUL/EXMPL1/
-----DSS--- ZREAD Unit  72; Vers.   2:  //BEND-FTRA/FLOW-PNLTY_EDT//AUG//
-----DSS---ZWRITE Unit  73; Vers.   3:  //BENDR-FTRA/FLOW-PNLTY_EDT//AUG/EXMPL1/
-----DSS--- ZREAD Unit  72; Vers.   2:  //BEND-FTRA/FLOW-PNLTY_EDT//SEP//
-----DSS---ZWRITE Unit  73; Vers.   3:  //BENDR-FTRA/FLOW-PNLTY_EDT//SEP/EXMPL1/
-----DSS--- ZREAD Unit  72; Vers.   2:  //BEND-FTRA/FLOW-PNLTY_EDT//OCT//
-----DSS---ZWRITE Unit  73; Vers.   3:  //BENDR-FTRA/FLOW-PNLTY_EDT//OCT/EXMPL1/
-----DSS--- ZREAD Unit  72; Vers.   2:  //BEND-FTRA/FLOW-PNLTY_EDT//JAN//
-----DSS---ZWRITE Unit  73; Vers.   3:  //BENDR-FTRA/FLOW-PNLTY_EDT//NOV/EXMPL1/
-----DSS---ZWRITE Unit  73; Vers.   3:  //BENDR-FTRA/FLOW-PNLTY_EDT//DEC/EXMPL1/
-----DSS--- ZREAD Unit  71; Vers.   1:  //FTRA/FLOW_LOC/01JAN1960/1MON//

NUMNEG:          2         SUMNEG:      -52.000          SUMNEG:     -0.520000E+02
Path: //FTRA/FLOW_LOC//1MON//


-----DSS--- ZREAD Unit  72; Vers.   1:  //FTRA/STOR-PNLTY_EDT//JAN//
-----DSS---ZWRITE Unit  73; Vers.   3:  //FTRA/STOR-PNLTY_EDT//JAN/EXMPL1/
-----DSS---ZWRITE Unit  73; Vers.   3:  //FTRA/STOR-PNLTY_EDT//FEB/EXMPL1/
-----DSS---ZWRITE Unit  73; Vers.   3:  //FTRA/STOR-PNLTY_EDT//MAR/EXMPL1/
-----DSS--- ZREAD Unit  72; Vers.   1:  //FTRA/STOR-PNLTY_EDT//APR//
-----DSS---ZWRITE Unit  73; Vers.   3:  //FTRA/STOR-PNLTY_EDT//APR/EXMPL1/
-----DSS--- ZREAD Unit  72; Vers.   1:  //FTRA/STOR-PNLTY_EDT//MAY//
-----DSS---ZWRITE Unit  73; Vers.   3:  //FTRA/STOR-PNLTY_EDT//MAY/EXMPL1/
-----DSS--- ZREAD Unit  72; Vers.   1:  //FTRA/STOR-PNLTY_EDT//JUN//
-----DSS---ZWRITE Unit  73; Vers.   3:  //FTRA/STOR-PNLTY_EDT//JUN/EXMPL1/
-----DSS--- ZREAD Unit  72; Vers.   1:  //FTRA/STOR-PNLTY_EDT//JUN//
-----DSS---ZWRITE Unit  73; Vers.   3:  //FTRA/STOR-PNLTY_EDT//JUL/EXMPL1/
-----DSS--- ZREAD Unit  72; Vers.   1:  //FTRA/STOR-PNLTY_EDT//AUG//
-----DSS---ZWRITE Unit  73; Vers.   3:  //FTRA/STOR-PNLTY_EDT//AUG/EXMPL1/
-----DSS--- ZREAD Unit  72; Vers.   1:  //FTRA/STOR-PNLTY_EDT//SEP//
-----DSS---ZWRITE Unit  73; Vers.   3:  //FTRA/STOR-PNLTY_EDT//SEP/EXMPL1/
-----DSS--- ZREAD Unit  72; Vers.   1:  //FTRA/STOR-PNLTY_EDT//MAY//
-----DSS---ZWRITE Unit  73; Vers.   3:  //FTRA/STOR-PNLTY_EDT//OCT/EXMPL1/
-----DSS--- ZREAD Unit  72; Vers.   1:  //FTRA/STOR-PNLTY_EDT//NOV//
-----DSS---ZWRITE Unit  73; Vers.   3:  //FTRA/STOR-PNLTY_EDT//NOV/EXMPL1/
-----DSS--- ZREAD Unit  72; Vers.   1:  //FTRA/STOR-PNLTY_EDT//JAN//
-----DSS---ZWRITE Unit  73; Vers.   3:  //FTRA/STOR-PNLTY_EDT//DEC/EXMPL1/
-----DSS--- ZREAD Unit  71; Vers.   2:  //FTRA/EVAP_RATE/01JAN1960/1MON//
-----DSS--- ZREAD Unit  72; Vers.   2:  //FTRA-GAPT/FLOW-PNLTY_HPE_EDT////
-----DSS---ZWRITE Unit  73; Vers.   3:  //FTRA-FTRAR/FLOW-PNLTY_HPE_EDT//JAN/EXMPL1/
-----DSS---ZWRITE Unit  73; Vers.   3:  //FTRA-FTRAR/FLOW-PNLTY_HPE_EDT//FEB/EXMPL1/
-----DSS---ZWRITE Unit  73; Vers.   3:  //FTRA-FTRAR/FLOW-PNLTY_HPE_EDT//MAR/EXMPL1/
-----DSS---ZWRITE Unit  73; Vers.   3:  //FTRA-FTRAR/FLOW-PNLTY_HPE_EDT//APR/EXMPL1/
-----DSS---ZWRITE Unit  73; Vers.   3:  //FTRA-FTRAR/FLOW-PNLTY_HPE_EDT//MAY/EXMPL1/
-----DSS---ZWRITE Unit  73; Vers.   3:  //FTRA-FTRAR/FLOW-PNLTY_HPE_EDT//JUN/EXMPL1/
-----DSS---ZWRITE Unit  73; Vers.   3:  //FTRA-FTRAR/FLOW-PNLTY_HPE_EDT//JUL/EXMPL1/
-----DSS---ZWRITE Unit  73; Vers.   3:  //FTRA-FTRAR/FLOW-PNLTY_HPE_EDT//AUG/EXMPL1/
-----DSS---ZWRITE Unit  73; Vers.   3:  //FTRA-FTRAR/FLOW-PNLTY_HPE_EDT//SEP/EXMPL1/
-----DSS---ZWRITE Unit  73; Vers.   3:  //FTRA-FTRAR/FLOW-PNLTY_HPE_EDT//OCT/EXMPL1/
-----DSS---ZWRITE Unit  73; Vers.   3:  //FTRA-FTRAR/FLOW-PNLTY_HPE_EDT//NOV/EXMPL1/
-----DSS---ZWRITE Unit  73; Vers.   3:  //FTRA-FTRAR/FLOW-PNLTY_HPE_EDT//DEC/EXMPL1/
-----DSS--- ZREAD Unit  72; Vers.   2:  //FTRA-GAPT/FLOW-PNLTY_EDT//JAN//
-----DSS---ZWRITE Unit  73; Vers.   3:  //FTRAR-GAPT/FLOW-PNLTY_EDT//JAN/EXMPL1/
-----DSS---ZWRITE Unit  73; Vers.   3:  //FTRAR-GAPT/FLOW-PNLTY_EDT//FEB/EXMPL1/
-----DSS---ZWRITE Unit  73; Vers.   3:  //FTRAR-GAPT/FLOW-PNLTY_EDT//MAR/EXMPL1/
-----DSS--- ZREAD Unit  72; Vers.   2:  //FTRA-GAPT/FLOW-PNLTY_EDT//APR//
-----DSS---ZWRITE Unit  73; Vers.   3:  //FTRAR-GAPT/FLOW-PNLTY_EDT//APR/EXMPL1/
-----DSS--- ZREAD Unit  72; Vers.   2:  //FTRA-GAPT/FLOW-PNLTY_EDT//JUN//
-----DSS---ZWRITE Unit  73; Vers.   3:  //FTRAR-GAPT/FLOW-PNLTY_EDT//MAY/EXMPL1/
```

```
-----DSS---ZWRITE Unit  73; Vers.   3:  //FTRAR-GAPT/FLOW-PNLTY_EDT//JUN/EXMPL1/
-----DSS--- ZREAD Unit  72; Vers.   2:  //FTRA-GAPT/FLOW-PNLTY_EDT//JUL//
-----DSS---ZWRITE Unit  73; Vers.   3:  //FTRAR-GAPT/FLOW-PNLTY_EDT//JUL/EXMPL1/
-----DSS--- ZREAD Unit  72; Vers.   2:  //FTRA-GAPT/FLOW-PNLTY_EDT//AUG//
-----DSS---ZWRITE Unit  73; Vers.   3:  //FTRAR-GAPT/FLOW-PNLTY_EDT//AUG/EXMPL1/
-----DSS--- ZREAD Unit  72; Vers.   2:  //FTRA-GAPT/FLOW-PNLTY_EDT//SEP//
-----DSS---ZWRITE Unit  73; Vers.   3:  //FTRAR-GAPT/FLOW-PNLTY_EDT//SEP/EXMPL1/
-----DSS--- ZREAD Unit  72; Vers.   2:  //FTRA-GAPT/FLOW-PNLTY_EDT//OCT//
-----DSS---ZWRITE Unit  73; Vers.   3:  //FTRAR-GAPT/FLOW-PNLTY_EDT//OCT/EXMPL1/
-----DSS--- ZREAD Unit  72; Vers.   2:  //FTRA-GAPT/FLOW-PNLTY_EDT//JAN//
-----DSS---ZWRITE Unit  73; Vers.   3:  //FTRAR-GAPT/FLOW-PNLTY_EDT//NOV/EXMPL1/
-----DSS---ZWRITE Unit  73; Vers.   3:  //FTRAR-GAPT/FLOW-PNLTY_EDT//DEC/EXMPL1/
-----DSS--- ZREAD Unit  71; Vers.   1:  //GAPT/FLOW_LOC/01JAN1960/1MON//
-----DSS--- ZREAD Unit  72; Vers.   1:  //GAPT/STOR-PNLTY_EDT//JAN//
-----DSS---ZWRITE Unit  73; Vers.   3:  //GAPT/STOR-PNLTY_EDT//JAN/EXMPL1/
-----DSS---ZWRITE Unit  73; Vers.   3:  //GAPT/STOR-PNLTY_EDT//FEB/EXMPL1/
-----DSS--- ZREAD Unit  72; Vers.   1:  //GAPT/STOR-PNLTY_EDT//MAR//
-----DSS---ZWRITE Unit  73; Vers.   3:  //GAPT/STOR-PNLTY_EDT//MAR/EXMPL1/
-----DSS--- ZREAD Unit  72; Vers.   1:  //GAPT/STOR-PNLTY_EDT//APR//
-----DSS---ZWRITE Unit  73; Vers.   3:  //GAPT/STOR-PNLTY_EDT//APR/EXMPL1/
-----DSS--- ZREAD Unit  72; Vers.   1:  //GAPT/STOR-PNLTY_EDT//MAY//
-----DSS---ZWRITE Unit  73; Vers.   3:  //GAPT/STOR-PNLTY_EDT//MAY/EXMPL1/
-----DSS---ZWRITE Unit  73; Vers.   3:  //GAPT/STOR-PNLTY_EDT//JUN/EXMPL1/
-----DSS--- ZREAD Unit  72; Vers.   1:  //GAPT/STOR-PNLTY_EDT//JUL//
-----DSS---ZWRITE Unit  73; Vers.   3:  //GAPT/STOR-PNLTY_EDT//JUL/EXMPL1/
-----DSS--- ZREAD Unit  72; Vers.   1:  //GAPT/STOR-PNLTY_EDT//AUG//
-----DSS---ZWRITE Unit  73; Vers.   3:  //GAPT/STOR-PNLTY_EDT//AUG/EXMPL1/
-----DSS--- ZREAD Unit  72; Vers.   1:  //GAPT/STOR-PNLTY_EDT//MAR//
-----DSS---ZWRITE Unit  73; Vers.   3:  //GAPT/STOR-PNLTY_EDT//SEP/EXMPL1/
-----DSS--- ZREAD Unit  72; Vers.   1:  //GAPT/STOR-PNLTY_EDT//JAN//
-----DSS---ZWRITE Unit  73; Vers.   3:  //GAPT/STOR-PNLTY_EDT//OCT/EXMPL1/
-----DSS---ZWRITE Unit  73; Vers.   3:  //GAPT/STOR-PNLTY_EDT//NOV/EXMPL1/
-----DSS---ZWRITE Unit  73; Vers.   3:  //GAPT/STOR-PNLTY_EDT//DEC/EXMPL1/
-----DSS--- ZREAD Unit  71; Vers.   2:  //GAPT/EVAP_RATE/01JAN1960/1MON//
-----DSS--- ZREAD Unit  72; Vers.   2:  //GAPT-SUX/FLOW-PNLTY_HPE_EDT////
-----DSS---ZWRITE Unit  73; Vers.   3:  //GAPT-GAPTR/FLOW-PNLTY_HPE_EDT//JAN/EXMPL1/
-----DSS---ZWRITE Unit  73; Vers.   3:  //GAPT-GAPTR/FLOW-PNLTY_HPE_EDT//FEB/EXMPL1/
-----DSS---ZWRITE Unit  73; Vers.   3:  //GAPT-GAPTR/FLOW-PNLTY_HPE_EDT//MAR/EXMPL1/
-----DSS---ZWRITE Unit  73; Vers.   3:  //GAPT-GAPTR/FLOW-PNLTY_HPE_EDT//APR/EXMPL1/
-----DSS---ZWRITE Unit  73; Vers.   3:  //GAPT-GAPTR/FLOW-PNLTY_HPE_EDT//MAY/EXMPL1/
-----DSS---ZWRITE Unit  73; Vers.   3:  //GAPT-GAPTR/FLOW-PNLTY_HPE_EDT//JUN/EXMPL1/
-----DSS---ZWRITE Unit  73; Vers.   3:  //GAPT-GAPTR/FLOW-PNLTY_HPE_EDT//JUL/EXMPL1/
-----DSS---ZWRITE Unit  73; Vers.   3:  //GAPT-GAPTR/FLOW-PNLTY_HPE_EDT//AUG/EXMPL1/
-----DSS---ZWRITE Unit  73; Vers.   3:  //GAPT-GAPTR/FLOW-PNLTY_HPE_EDT//SEP/EXMPL1/
-----DSS---ZWRITE Unit  73; Vers.   3:  //GAPT-GAPTR/FLOW-PNLTY_HPE_EDT//OCT/EXMPL1/
-----DSS---ZWRITE Unit  73; Vers.   3:  //GAPT-GAPTR/FLOW-PNLTY_HPE_EDT//NOV/EXMPL1/
-----DSS---ZWRITE Unit  73; Vers.   3:  //GAPT-GAPTR/FLOW-PNLTY_HPE_EDT//DEC/EXMPL1/
-----DSS--- ZREAD Unit  72; Vers.   2:  //GAPT-SUX/FLOW-PNLTY_EDT//JAN//
-----DSS---ZWRITE Unit  73; Vers.   3:  //GAPTR-SUX/FLOW-PNLTY_EDT//JAN/EXMPL1/
-----DSS---ZWRITE Unit  73; Vers.   3:  //GAPTR-SUX/FLOW-PNLTY_EDT//FEB/EXMPL1/
-----DSS---ZWRITE Unit  73; Vers.   3:  //GAPTR-SUX/FLOW-PNLTY_EDT//MAR/EXMPL1/
-----DSS--- ZREAD Unit  72; Vers.   2:  //GAPT-SUX/FLOW-PNLTY_EDT//APR//
-----DSS---ZWRITE Unit  73; Vers.   3:  //GAPTR-SUX/FLOW-PNLTY_EDT//APR/EXMPL1/
-----DSS--- ZREAD Unit  72; Vers.   2:  //GAPT-SUX/FLOW-PNLTY_EDT//MAY//
-----DSS---ZWRITE Unit  73; Vers.   3:  //GAPTR-SUX/FLOW-PNLTY_EDT//MAY/EXMPL1/
-----DSS---ZWRITE Unit  73; Vers.   3:  //GAPTR-SUX/FLOW-PNLTY_EDT//JUN/EXMPL1/
-----DSS--- ZREAD Unit  72; Vers.   2:  //GAPT-SUX/FLOW-PNLTY_EDT//JUL//
-----DSS---ZWRITE Unit  73; Vers.   3:  //GAPTR-SUX/FLOW-PNLTY_EDT//JUL/EXMPL1/
-----DSS--- ZREAD Unit  72; Vers.   2:  //GAPT-SUX/FLOW-PNLTY_EDT//AUG//
-----DSS---ZWRITE Unit  73; Vers.   3:  //GAPTR-SUX/FLOW-PNLTY_EDT//AUG/EXMPL1/
-----DSS--- ZREAD Unit  72; Vers.   2:  //GAPT-SUX/FLOW-PNLTY_EDT//SEP//
-----DSS---ZWRITE Unit  73; Vers.   3:  //GAPTR-SUX/FLOW-PNLTY_EDT//SEP/EXMPL1/
-----DSS--- ZREAD Unit  72; Vers.   2:  //GAPT-SUX/FLOW-PNLTY_EDT//OCT//
-----DSS---ZWRITE Unit  73; Vers.   3:  //GAPTR-SUX/FLOW-PNLTY_EDT//OCT/EXMPL1/
-----DSS--- ZREAD Unit  72; Vers.   2:  //GAPT-SUX/FLOW-PNLTY_EDT//JAN//
-----DSS---ZWRITE Unit  73; Vers.   3:  //GAPTR-SUX/FLOW-PNLTY_EDT//NOV/EXMPL1/
-----DSS---ZWRITE Unit  73; Vers.   3:  //GAPTR-SUX/FLOW-PNLTY_EDT//DEC/EXMPL1/
-----DSS--- ZREAD Unit  71; Vers.   1:  //SUX/FLOW_LOC/01JAN1960/1MON//
-----DSS--- ZREAD Unit  72; Vers.   2:  //SUX-OMA/FLOW-PNLTY_EDT//JAN//
-----DSS---ZWRITE Unit  73; Vers.   3:  //SUX-OMA/FLOW-PNLTY_EDT//JAN/EXMPL1/
-----DSS---ZWRITE Unit  73; Vers.   3:  //SUX-OMA/FLOW-PNLTY_EDT//FEB/EXMPL1/
-----DSS---ZWRITE Unit  73; Vers.   3:  //SUX-OMA/FLOW-PNLTY_EDT//MAR/EXMPL1/
-----DSS--- ZREAD Unit  72; Vers.   2:  //SUX-OMA/FLOW-PNLTY_EDT//APR//
-----DSS---ZWRITE Unit  73; Vers.   3:  //SUX-OMA/FLOW-PNLTY_EDT//APR/EXMPL1/
-----DSS--- ZREAD Unit  72; Vers.   2:  //SUX-OMA/FLOW-PNLTY_EDT//MAY//
-----DSS---ZWRITE Unit  73; Vers.   3:  //SUX-OMA/FLOW-PNLTY_EDT//MAY/EXMPL1/
-----DSS--- ZREAD Unit  72; Vers.   2:  //SUX-OMA/FLOW-PNLTY_EDT//JUN//
-----DSS---ZWRITE Unit  73; Vers.   3:  //SUX-OMA/FLOW-PNLTY_EDT//JUN/EXMPL1/
-----DSS--- ZREAD Unit  72; Vers.   2:  //SUX-OMA/FLOW-PNLTY_EDT//JUL//
-----DSS---ZWRITE Unit  73; Vers.   3:  //SUX-OMA/FLOW-PNLTY_EDT//JUL/EXMPL1/
-----DSS---ZWRITE Unit  73; Vers.   3:  //SUX-OMA/FLOW-PNLTY_EDT//AUG/EXMPL1/
-----DSS--- ZREAD Unit  72; Vers.   2:  //SUX-OMA/FLOW-PNLTY_EDT//SEP//
-----DSS---ZWRITE Unit  73; Vers.   3:  //SUX-OMA/FLOW-PNLTY_EDT//SEP/EXMPL1/
-----DSS--- ZREAD Unit  72; Vers.   2:  //SUX-OMA/FLOW-PNLTY_EDT//OCT//
-----DSS---ZWRITE Unit  73; Vers.   3:  //SUX-OMA/FLOW-PNLTY_EDT//OCT/EXMPL1/
-----DSS--- ZREAD Unit  72; Vers.   2:  //SUX-OMA/FLOW-PNLTY_EDT//NOV//
-----DSS---ZWRITE Unit  73; Vers.   3:  //SUX-OMA/FLOW-PNLTY_EDT//NOV/EXMPL1/
-----DSS--- ZREAD Unit  72; Vers.   2:  //SUX-OMA/FLOW-PNLTY_EDT//DEC//
-----DSS---ZWRITE Unit  73; Vers.   3:  //SUX-OMA/FLOW-PNLTY_EDT//DEC/EXMPL1/
-----DSS--- ZREAD Unit  71; Vers.   1:  //OMA/FLOW_LOC/01JAN1960/1MON//
-----DSS--- ZREAD Unit  72; Vers.   2:  //OMA-NONE/FLOW-PNLTY_EDT//JAN//
-----DSS---ZWRITE Unit  73; Vers.   3:  //OMA-NONE/FLOW-PNLTY_EDT//JAN/EXMPL1/
```

```
-----DSS---ZWRITE Unit  73; Vers.    3:  //OMA-NCNE/FLOW-PNLTY_EDT//FEB/EXMPL1/
-----DSS---ZWRITE Unit  73; Vers.    3:  //OMA-NCNE/FLOW-PNLTY_EDT//MAR/EXMPL1/
-----DSS--- ZREAD Unit  72; Vers.    2:  //OMA-NCNE/FLOW-PNLTY_EDT//APR//
-----DSS---ZWRITE Unit  73; Vers.    3:  //OMA-NCNE/FLOW-PNLTY_EDT//APR/EXMPL1/
-----DSS---ZWRITE Unit  73; Vers.    3:  //OMA-NCNE/FLOW-PNLTY_EDT//MAY/EXMPL1/
-----DSS---ZWRITE Unit  73; Vers.    3:  //OMA-NCNE/FLOW-PNLTY_EDT//JUN/EXMPL1/
-----DSS---ZWRITE Unit  73; Vers.    3:  //OMA-NCNE/FLOW-PNLTY_EDT//JUL/EXMPL1/
-----DSS---ZWRITE Unit  73; Vers.    3:  //OMA-NCNE/FLOW-PNLTY_EDT//AUG/EXMPL1/
-----DSS---ZWRITE Unit  73; Vers.    3:  //OMA-NCNE/FLOW-PNLTY_EDT//SEP/EXMPL1/
-----DSS---ZWRITE Unit  73; Vers.    3:  //OMA-NCNE/FLOW-PNLTY_EDT//OCT/EXMPL1/
-----DSS---ZWRITE Unit  73; Vers.    3:  //OMA-NCNE/FLOW-PNLTY_EDT//NOV/EXMPL1/
-----DSS--- ZREAD Unit  72; Vers.    2:  //OMA-NCNE/FLOW-PNLTY_EDT//JAN//
-----DSS---ZWRITE Unit  73; Vers.    3:  //OMA-NCNE/FLOW-PNLTY_EDT//DEC/EXMPL1/
-----DSS--- ZREAD Unit  71; Vers.    1:  //NCNE/FLOW_LOC/01JAN1960/1MON//
-----DSS--- ZREAD Unit  72; Vers.    2:  //NCNE-MKC/FLOW-PNLTY_EDT//JAN//
-----DSS---ZWRITE Unit  73; Vers.    3:  //NCNE-MKC/FLOW-PNLTY_EDT//JAN/EXMPL1/
-----DSS---ZWRITE Unit  73; Vers.    3:  //NCNE-MKC/FLOW-PNLTY_EDT//FEB/EXMPL1/
-----DSS---ZWRITE Unit  73; Vers.    3:  //NCNE-MKC/FLOW-PNLTY_EDT//MAR/EXMPL1/
-----DSS--- ZREAD Unit  72; Vers.    2:  //NCNE-MKC/FLOW-PNLTY_EDT//APR//
-----DSS---ZWRITE Unit  73; Vers.    3:  //NCNE-MKC/FLOW-PNLTY_EDT//APR/EXMPL1/
-----DSS---ZWRITE Unit  73; Vers.    3:  //NCNE-MKC/FLOW-PNLTY_EDT//MAY/EXMPL1/
-----DSS---ZWRITE Unit  73; Vers.    3:  //NCNE-MKC/FLOW-PNLTY_EDT//JUN/EXMPL1/
-----DSS--- ZREAD Unit  72; Vers.    2:  //NCNE-MKC/FLOW-PNLTY_EDT//JUL//
-----DSS---ZWRITE Unit  73; Vers.    3:  //NCNE-MKC/FLOW-PNLTY_EDT//JUL/EXMPL1/
-----DSS---ZWRITE Unit  73; Vers.    3:  //NCNE-MKC/FLOW-PNLTY_EDT//AUG/EXMPL1/
-----DSS---ZWRITE Unit  73; Vers.    3:  //NCNE-MKC/FLOW-PNLTY_EDT//SEP/EXMPL1/
-----DSS--- ZREAD Unit  72; Vers.    2:  //NCNE-MKC/FLOW-PNLTY_EDT//OCT//
-----DSS---ZWRITE Unit  73; Vers.    3:  //NCNE-MKC/FLOW-PNLTY_EDT//OCT/EXMPL1/
-----DSS---ZWRITE Unit  73; Vers.    3:  //NCNE-MKC/FLOW-PNLTY_EDT//NOV/EXMPL1/
-----DSS--- ZREAD Unit  72; Vers.    2:  //NCNE-MKC/FLOW-PNLTY_EDT//DEC//
-----DSS---ZWRITE Unit  73; Vers.    3:  //NCNE-MKC/FLOW-PNLTY_EDT//DEC/EXMPL1/
-----DSS--- ZREAD Unit  71; Vers.    1:  //MKC/FLOW_LOC/01JAN1960/1MON//
-----DSS--- ZREAD Unit  72; Vers.    2:  //MKC-BNMO/FLOW-PNLTY_EDT//JAN//
-----DSS---ZWRITE Unit  73; Vers.    3:  //MKC-BNMO/FLOW-PNLTY_EDT//JAN/EXMPL1/
-----DSS---ZWRITE Unit  73; Vers.    3:  //MKC-BNMO/FLOW-PNLTY_EDT//FEB/EXMPL1/
-----DSS--- ZREAD Unit  72; Vers.    2:  //MKC-BNMO/FLOW-PNLTY_EDT//MAR//
-----DSS---ZWRITE Unit  73; Vers.    3:  //MKC-BNMO/FLOW-PNLTY_EDT//MAR/EXMPL1/
-----DSS--- ZREAD Unit  72; Vers.    2:  //MKC-BNMO/FLOW-PNLTY_EDT//APR//
-----DSS---ZWRITE Unit  73; Vers.    3:  //MKC-BNMO/FLOW-PNLTY_EDT//APR/EXMPL1/
-----DSS--- ZREAD Unit  72; Vers.    2:  //MKC-BNMO/FLOW-PNLTY_EDT//MAY//
-----DSS---ZWRITE Unit  73; Vers.    3:  //MKC-BNMO/FLOW-PNLTY_EDT//MAY/EXMPL1/
-----DSS--- ZREAD Unit  72; Vers.    2:  //MKC-BNMO/FLOW-PNLTY_EDT//JUN//
-----DSS---ZWRITE Unit  73; Vers.    3:  //MKC-BNMO/FLOW-PNLTY_EDT//JUN/EXMPL1/
-----DSS--- ZREAD Unit  72; Vers.    2:  //MKC-BNMO/FLOW-PNLTY_EDT//JUL//
-----DSS---ZWRITE Unit  73; Vers.    3:  //MKC-BNMO/FLOW-PNLTY_EDT//JUL/EXMPL1/
-----DSS---ZWRITE Unit  73; Vers.    3:  //MKC-BNMO/FLOW-PNLTY_EDT//AUG/EXMPL1/
-----DSS--- ZREAD Unit  72; Vers.    2:  //MKC-BNMO/FLOW-PNLTY_EDT//SEP//
-----DSS---ZWRITE Unit  73; Vers.    3:  //MKC-BNMO/FLOW-PNLTY_EDT//SEP/EXMPL1/
-----DSS--- ZREAD Unit  72; Vers.    2:  //MKC-BNMO/FLOW-PNLTY_EDT//OCT//
-----DSS---ZWRITE Unit  73; Vers.    3:  //MKC-BNMO/FLOW-PNLTY_EDT//OCT/EXMPL1/
-----DSS--- ZREAD Unit  72; Vers.    2:  //MKC-BNMO/FLOW-PNLTY_EDT//NOV//
-----DSS---ZWRITE Unit  73; Vers.    3:  //MKC-BNMO/FLOW-PNLTY_EDT//NOV/EXMPL1/
-----DSS--- ZREAD Unit  72; Vers.    2:  //MKC-BNMO/FLOW-PNLTY_EDT//DEC//
-----DSS---ZWRITE Unit  73; Vers.    3:  //MKC-BNMO/FLOW-PNLTY_EDT//DEC/EXMPL1/
-----DSS--- ZREAD Unit  71; Vers.    1:  //BNMO/FLOW_LOC/01JAN1960/1MON//
-----DSS--- ZREAD Unit  72; Vers.    2:  //BNMO-HEMO/FLOW-PNLTY_EDT//JAN//
-----DSS---ZWRITE Unit  73; Vers.    3:  //BNMO-HEMO/FLOW-PNLTY_EDT//JAN/EXMPL1/
-----DSS---ZWRITE Unit  73; Vers.    3:  //BNMO-HEMO/FLOW-PNLTY_EDT//FEB/EXMPL1/
-----DSS---ZWRITE Unit  73; Vers.    3:  //BNMO-HEMO/FLOW-PNLTY_EDT//MAR/EXMPL1/
-----DSS--- ZREAD Unit  72; Vers.    2:  //BNMO-HEMO/FLOW-PNLTY_EDT//APR//
-----DSS---ZWRITE Unit  73; Vers.    3:  //BNMO-HEMO/FLOW-PNLTY_EDT//APR/EXMPL1/
-----DSS--- ZREAD Unit  72; Vers.    2:  //BNMO-HEMO/FLOW-PNLTY_EDT//MAY//
-----DSS---ZWRITE Unit  73; Vers.    3:  //BNMO-HEMO/FLOW-PNLTY_EDT//MAY/EXMPL1/
-----DSS--- ZREAD Unit  72; Vers.    2:  //BNMO-HEMO/FLOW-PNLTY_EDT//JUN//
-----DSS---ZWRITE Unit  73; Vers.    3:  //BNMO-HEMO/FLOW-PNLTY_EDT//JUN/EXMPL1/
-----DSS--- ZREAD Unit  72; Vers.    2:  //BNMO-HEMO/FLOW-PNLTY_EDT//JUL//
-----DSS---ZWRITE Unit  73; Vers.    3:  //BNMO-HEMO/FLOW-PNLTY_EDT//JUL/EXMPL1/
-----DSS---ZWRITE Unit  73; Vers.    3:  //BNMO-HEMO/FLOW-PNLTY_EDT//AUG/EXMPL1/
-----DSS---ZWRITE Unit  73; Vers.    3:  //BNMO-HEMO/FLOW-PNLTY_EDT//SEP/EXMPL1/
-----DSS--- ZREAD Unit  72; Vers.    2:  //BNMO-HEMO/FLOW-PNLTY_EDT//OCT//
-----DSS---ZWRITE Unit  73; Vers.    3:  //BNMO-HEMO/FLOW-PNLTY_EDT//OCT/EXMPL1/
-----DSS--- ZREAD Unit  72; Vers.    2:  //BNMO-HEMO/FLOW-PNLTY_EDT//NOV//
-----DSS---ZWRITE Unit  73; Vers.    3:  //BNMO-HEMO/FLOW-PNLTY_EDT//NOV/EXMPL1/
-----DSS---ZWRITE Unit  73; Vers.    3:  //BNMO-HEMO/FLOW-PNLTY_EDT//DEC/EXMPL1/
-----DSS--- ZREAD Unit  71; Vers.    1:  //HEMO/FLOW_LOC/01JAN1960/1MON//
-----DSS--- ZREAD Unit  72; Vers.    2:  //HEMO-STL/FLOW-PNLTY_EDT//JAN//
-----DSS---ZWRITE Unit  73; Vers.    3:  //HEMO-STL/FLOW-PNLTY_EDT//JAN/EXMPL1/
-----DSS---ZWRITE Unit  73; Vers.    3:  //HEMO-STL/FLOW-PNLTY_EDT//FEB/EXMPL1/
-----DSS---ZWRITE Unit  73; Vers.    3:  //HEMO-STL/FLOW-PNLTY_EDT//MAR/EXMPL1/
-----DSS--- ZREAD Unit  72; Vers.    2:  //HEMO-STL/FLOW-PNLTY_EDT//APR//
-----DSS---ZWRITE Unit  73; Vers.    3:  //HEMO-STL/FLOW-PNLTY_EDT//APR/EXMPL1/
-----DSS--- ZREAD Unit  72; Vers.    2:  //HEMO-STL/FLOW-PNLTY_EDT//MAY//
-----DSS---ZWRITE Unit  73; Vers.    3:  //HEMO-STL/FLOW-PNLTY_EDT//MAY/EXMPL1/
-----DSS--- ZREAD Unit  72; Vers.    2:  //HEMO-STL/FLOW-PNLTY_EDT//JUN//
-----DSS---ZWRITE Unit  73; Vers.    3:  //HEMO-STL/FLOW-PNLTY_EDT//JUN/EXMPL1/
-----DSS--- ZREAD Unit  72; Vers.    2:  //HEMO-STL/FLOW-PNLTY_EDT//JUL//
-----DSS---ZWRITE Unit  73; Vers.    3:  //HEMO-STL/FLOW-PNLTY_EDT//JUL/EXMPL1/
-----DSS---ZWRITE Unit  73; Vers.    3:  //HEMO-STL/FLOW-PNLTY_EDT//AUG/EXMPL1/
-----DSS---ZWRITE Unit  73; Vers.    3:  //HEMO-STL/FLOW-PNLTY_EDT//SEP/EXMPL1/
-----DSS--- ZREAD Unit  72; Vers.    2:  //HEMO-STL/FLOW-PNLTY_EDT//OCT//
-----DSS---ZWRITE Unit  73; Vers.    3:  //HEMO-STL/FLOW-PNLTY_EDT//OCT/EXMPL1/
-----DSS--- ZREAD Unit  72; Vers.    2:  //HEMO-STL/FLOW-PNLTY_EDT//NOV//
```

```
-----DSS---ZWRITE Unit  73; Vers.    3:  //HEMO-STL/FLOW-PNLTY_EDT//NOV/EXMPL1/
-----DSS---ZWRITE Unit  73; Vers.    3:  //HEMO-STL/FLOW-PNLTY_EDT//DEC/EXMPL1/
-----DSS--- ZREAD Unit  72; Vers.    2:  //STL-S_SINK/FLOW-PNLTY_EDT//JAN//
-----DSS---ZWRITE Unit  73; Vers.    3:  //STL-S_SINK/FLOW-PNLTY_EDT//JAN/EXMPL1/
-----DSS---ZWRITE Unit  73; Vers.    3:  //STL-S_SINK/FLOW-PNLTY_EDT//FEB/EXMPL1/
-----DSS---ZWRITE Unit  73; Vers.    3:  //STL-S_SINK/FLOW-PNLTY_EDT//MAR/EXMPL1/
-----DSS---ZWRITE Unit  73; Vers.    3:  //STL-S_SINK/FLOW-PNLTY_EDT//APR/EXMPL1/
-----DSS---ZWRITE Unit  73; Vers.    3:  //STL-S_SINK/FLOW-PNLTY_EDT//MAY/EXMPL1/
-----DSS---ZWRITE Unit  73; Vers.    3:  //STL-S_SINK/FLOW-PNLTY_EDT//JUN/EXMPL1/
-----DSS---ZWRITE Unit  73; Vers.    3:  //STL-S_SINK/FLOW-PNLTY_EDT//JUL/EXMPL1/
-----DSS---ZWRITE Unit  73; Vers.    3:  //STL-S_SINK/FLOW-PNLTY_EDT//AUG/EXMPL1/
-----DSS---ZWRITE Unit  73; Vers.    3:  //STL-S_SINK/FLOW-PNLTY_EDT//SEP/EXMPL1/
-----DSS---ZWRITE Unit  73; Vers.    3:  //STL-S_SINK/FLOW-PNLTY_EDT//OCT/EXMPL1/
-----DSS---ZWRITE Unit  73; Vers.    3:  //STL-S_SINK/FLOW-PNLTY_EDT//NOV/EXMPL1/
-----DSS---ZWRITE Unit  73; Vers.    3:  //STL-S_SINK/FLOW-PNLTY_EDT//DEC/EXMPL1/
```

43 Links in this system

#	From	To	Type	Mult	Cost	Sum Neg	Num Neg	Description
1	S_SOURCE	S_SINK	DIVR	1.00000	0.00	0.0	0	Link to balance flow throughout computations
2	FTPK	FTPK	STO1	1.00000	0.00	0.0	0	
3	GARR	GARR	STO1	1.00000	0.00	0.0	0	
4	OAHE	OAHE	STO1	1.00000	0.00	0.0	0	
5	BEND	BEND	STO1	1.00000	0.00	0.0	0	
6	FTRA	FTRA	STO1	1.00000	0.00	0.0	0	
7	GAPT	GAPT	STO1	1.00000	0.00	0.0	0	
8	S_SOURCE	FTPK	INFL	1.00000	0.00	0.0	0	Inflow into Fort Peck Reservoir
9	FTPK	FTPK	RSTO	1.00000	0.00	0.0	0	Storage in Fort Peck Reservoir
10	FTPK	FTPKR	HREL	1.00000	0.00	0.0	0	Hydropower release from Ft. Peck
11	FTPKR	GARR	RREL	1.00000	0.00	0.0	0	Release from Fort Peck Reservoir to Garrison
12	S_SOURCE	GARR	INFL	1.00000	0.00	0.0	0	Inflow into Garrison Reservoir
13	GARR	GARR	RSTO	1.00000	0.00	0.0	0	Storage in Garrison Reservoir
14	GARR	GARRR	HREL	1.00000	0.00	0.0	0	Release from Garrison Reservoir to Oahe (energy)
15	GARRR	OAHE	RREL	1.00000	0.00	0.0	0	Release from Garrison Reservoir to Oahe
16	S_SOURCE	OAHE	INFL	1.00000	0.00	-266.0	3	Inflow into Oahe Reservoir
17	OAHE	OAHE	RSTO	1.00000	0.00	0.0	0	Storage in Oahe Reservoir
18	OAHE	OAHER	HREL	1.00000	0.00	0.0	0	Release from Oahe Reservoir to Big Bend Reservoir (energy)
19	OAHER	BEND	RREL	1.00000	0.00	0.0	0	Release from Oahe Reservoir to Big Bend Reservoir
20	BEND	BEND	RSTO	1.00000	0.00	0.0	0	Storage in Big Bend Reservoir
21	BEND	BENDR	HREL	1.00000	0.00	0.0	0	Release from Big Bend Reservoir to Fort Randall Reservoir (energy)
22	BENDR	FTRA	RREL	1.00000	0.00	0.0	0	Release from Big Bend Reservoir to Fort Randall Reservoir
23	S_SOURCE	FTRA	INFL	1.00000	0.00	-52.0	2	Inflow into Fort Randall Reservoir
24	FTRA	FTRA	RSTO	1.00000	0.00	0.0	0	Storage in Fort Randall Reservoir
25	FTRA	FTRAR	HREL	1.00000	0.00	0.0	0	Release from Fort Randal Reservoir to Gavins Point Reservoir (energy)
26	FTRAR	GAPT	RREL	1.00000	0.00	0.0	0	Release from Fort Randal Reservoir to Gavins Point Reservoir
27	S_SOURCE	GAPT	INFL	1.00000	0.00	0.0	0	Inflow into Gavins Point Reservoir
28	GAPT	GAPT	RSTO	1.00000	0.00	0.0	0	Storage in Gavins Point Reservoir
29	GAPT	GAPTR	HREL	1.00000	0.00	0.0	0	Release from Gavins Point to Sioux City (energy)
30	GAPTR	SUX	RREL	1.00000	0.00	0.0	0	Release from Gavins Point to Sioux City
31	S_SOURCE	SUX	INFL	1.00000	0.00	0.0	0	Inflow to Sioux City
32	SUX	OMA	CHAN	1.00000	0.00	0.0	0	Channel from Sioux City to Omaha
33	S_SOURCE	OMA	INFL	1.00000	0.00	0.0	0	Inflow to Omaha
34	OMA	NCNE	CHAN	1.00000	0.00	0.0	0	Channel from Omaha to Nebraska City
35	S_SOURCE	NCNE	INFL	1.00000	0.00	0.0	0	Inflow to Nebraska City
36	NCNE	MKC	CHAN	1.00000	0.00	0.0	0	Channel from Nebraska City to Kansas City
37	S_SOURCE	MKC	INFL	1.00000	0.00	0.0	0	Inflow to Kansas City
38	MKC	BNMO	CHAN	1.00000	0.00	0.0	0	Channel from Kansas City to St. Louis
39	S_SOURCE	BNMO	INFL	1.00000	0.00	0.0	0	Inflow to Boonville
40	BNMO	HEMO	CHAN	1.00000	0.00	0.0	0	Channel from Boonville to Hermann
41	S_SOURCE	HEMO	INFL	1.00000	0.00	0.0	0	Inflow to Hermann
42	HEMO	STL	CHAN	1.00000	0.00	0.0	0	Channel from Hermann to St. Louis (Sink for now)
43	STL	S_SINK	CHAN	1.00000	0.00	0.0	0	Channel from St. Louis to the Super Sink

{Print parameters and network solver matrix before solution}

```
Parameters associated with the Network Solver
_______________________________________________

  SOURCE     SINK    NARC   NODES      IFS    IROOT     NDEG     NLOP     ITER    NPRIT
       1        2    1071     230        0        0        0        0        0        0

  IPRINT
       0

          OUTFLO            FLONET            CSTNOW            TOTCST               EPS
    126749.00000           0.00000           0.00000           0.00000           0.00001

             BIG              SICH             TIMAX              TIME
 1000000.00000****************           0.00000           0.00001

          OUTFLO            FLONET            CSTNOW            TOTCST               EPS
 0.1267490E+06     0.0000000E+00     0.0000000E+00     0.0000000E+00     0.1000000E-04

             BIG              SICH             TIMAX              TIME
 0.1000000E+07     0.1000000E+11     0.0000000E+00     0.1000000E-04
```

Arc #	IARC	JARC	LOWER	UPPER	COST	AMP	FLOW	COST
1	1	2	0.00	126749.00	0.0000000E+00	1.0000	0.00	0.000000
2	1	3	13900.00	13900.00	0.0000000E+00	1.0000	0.00	0.000000
3	1	5	15605.00	15605.00	0.0000000E+00	1.0000	0.00	0.000000
4	1	7	17739.00	17739.00	0.0000000E+00	1.0000	0.00	0.000000
5	1	9	1730.00	1730.00	0.0000000E+00	1.0000	0.00	0.000000
6	1	11	3473.00	3473.00	0.0000000E+00	1.0000	0.00	0.000000
7	1	13	357.00	357.00	0.0000000E+00	1.0000	0.00	0.000000
8	1	3	722.00	722.00	0.0000000E+00	1.0000	0.00	0.000000
9	1	22	1197.00	1197.00	0.0000000E+00	1.0000	0.00	0.000000
10	1	41	1429.00	1429.00	0.0000000E+00	1.0000	0.00	0.000000
11	1	60	1790.00	1790.00	0.0000000E+00	1.0000	0.00	0.000000
12	1	79	1323.00	1323.00	0.0000000E+00	1.0000	0.00	0.000000
13	1	98	667.00	667.00	0.0000000E+00	1.0000	0.00	0.000000
14	1	117	714.00	714.00	0.0000000E+00	1.0000	0.00	0.000000
15	1	136	1064.00	1064.00	0.0000000E+00	1.0000	0.00	0.000000
16	1	155	819.00	819.00	0.0000000E+00	1.0000	0.00	0.000000
17	1	174	497.00	497.00	0.0000000E+00	1.0000	0.00	0.000000
18	1	193	465.00	465.00	0.0000000E+00	1.0000	0.00	0.000000
19	1	212	445.00	445.00	0.0000000E+00	1.0000	0.00	0.000000
20	3	22	4211.00	10782.61	-0.2051292E+03	1.0000	0.00	-205.129200
21	3	22	0.00	4289.86	-0.1234102E+03	1.0000	0.00	-123.410200
22	3	22	0.00	1594.20	-0.1844921E+02	1.0000	0.00	-18.449210
23	3	22	0.00	1047.33	0.1383689E+03	1.0000	0.00	138.368900
24	22	41	4211.00	11130.43	-0.2382582E+03	1.0000	0.00	-238.258200
25	22	41	0.00	3884.06	-0.1191396E+03	1.0000	0.00	-119.139600
26	22	41	0.00	1565.22	0.5010889E+01	1.0000	0.00	5.010889
27	22	41	0.00	1134.29	0.1448219E+03	1.0000	0.00	144.821900
28	41	60	4211.00	11130.43	-0.2382582E+03	0.9983	0.00	-238.258200
29	41	60	0.00	3884.06	-0.1191396E+03	0.9983	0.00	-119.139600
30	41	60	0.00	1565.22	0.5010889E+01	0.9983	0.00	5.010889
31	41	60	0.00	1134.29	0.1448219E+03	0.9983	0.00	144.821900
32	60	79	4211.00	11130.43	-0.2382582E+03	0.9988	0.00	-238.258200
33	60	79	0.00	3884.06	-0.1191396E+03	0.9988	0.00	-119.139600
34	60	79	0.00	1565.22	0.5010889E+01	0.9988	0.00	5.010889
35	60	79	0.00	1134.29	0.1448219E+03	0.9988	0.00	144.821900
36	79	98	4211.00	11072.46	-0.3223391E+03	0.9954	0.00	-322.339100
37	79	98	0.00	3826.09	-0.1358066E+03	0.9954	0.00	-135.806600
38	79	98	0.00	1652.17	0.1780186E+02	0.9954	0.00	17.801860
39	79	98	0.00	1163.28	0.2009804E+03	0.9954	0.00	200.980400
40	98	117	4211.00	11130.43	-0.2382582E+03	0.9951	0.00	-238.258200
41	98	117	0.00	3884.06	-0.1191396E+03	0.9951	0.00	-119.139600
42	98	117	0.00	1565.22	0.5010889E+01	0.9951	0.00	5.010889
43	98	117	0.00	1134.29	0.1448219E+03	0.9951	0.00	144.821900
44	117	136	4211.00	11043.48	-0.2218894E+03	0.9954	0.00	-221.889400
45	117	136	0.00	3913.04	-0.1139978E+03	0.9954	0.00	-113.997800
46	117	136	0.00	1594.20	-0.8609623E+01	0.9954	0.00	-8.609623
47	117	136	0.00	1163.28	0.1333886E+03	0.9954	0.00	133.388600
48	136	155	4211.00	11130.43	-0.2382582E+03	0.9968	0.00	-238.258200
49	136	155	0.00	3884.06	-0.1191396E+03	0.9968	0.00	-119.139600
50	136	155	0.00	1565.22	0.5010889E+01	0.9968	0.00	5.010889
51	136	155	0.00	1134.29	0.1448219E+03	0.9968	0.00	144.821900
52	155	174	4211.00	11043.48	-0.2218894E+03	0.9971	0.00	-221.889400
53	155	174	0.00	3913.04	-0.1139978E+03	0.9971	0.00	-113.997800
54	155	174	0.00	1594.20	-0.8609623E+01	0.9971	0.00	-8.609623
55	155	174	0.00	1163.28	0.1333886E+03	0.9971	0.00	133.388600
56	174	193	4211.00	10956.52	-0.1916667E+03	0.9988	0.00	-191.666700
57	174	193	0.00	3768.12	-0.1019909E+03	0.9988	0.00	-101.990900
58	174	193	0.00	1913.04	-0.3587343E+02	0.9988	0.00	-35.873430
59	174	193	0.00	1076.32	0.1150001E+03	0.9988	0.00	115.000100
60	193	212	4211.00	10782.61	-0.2051292E+03	1.0000	0.00	-205.129200

{additional output deleted}

1038	78	2	0.00	9492.75	0.4750994E+03	1.0000	0.00	475.099400
1039	78	2	0.00	980760.90	0.2321544E+04	1.0000	0.00	2321.544000
1040	97	2	0.00	326.09	-0.1827529E+08	1.0000	0.00	**************
1041	97	2	0.00	9420.29	0.4246154E+01	1.0000	0.00	4.246154
1042	97	2	0.00	9492.75	0.4750994E+03	1.0000	0.00	475.099400
1043	97	2	0.00	980760.90	0.2321544E+04	1.0000	0.00	2321.544000
1044	116	2	0.00	326.09	-0.1827529E+08	1.0000	0.00	**************
1045	116	2	0.00	9420.29	0.4246154E+01	1.0000	0.00	4.246154
1046	116	2	0.00	9492.75	0.4750994E+03	1.0000	0.00	475.099400
1047	116	2	0.00	980760.90	0.2321544E+04	1.0000	0.00	2321.544000
1048	135	2	0.00	326.09	-0.1827529E+08	1.0000	0.00	**************
1049	135	2	0.00	9420.29	0.4246154E+01	1.0000	0.00	4.246154
1050	135	2	0.00	9492.75	0.4750994E+03	1.0000	0.00	475.099400
1051	135	2	0.00	980760.90	0.2321544E+04	1.0000	0.00	2321.544000
1052	154	2	0.00	326.09	-0.1827529E+08	1.0000	0.00	**************
[illegible]	154	2	0.00	9420.29	0.4246154E+01	1.0000	0.00	4.246154
[illegible]	[illegible]	2	0.00	9492.75	0.4750994E+03	1.0000	0.00	475.099400
[illegible]	[illegible]	2	0.00	980760.90	0.2321544E+04	1.0000	0.00	2321.544000
[illegible]	[illegible]	2	0.00	[illegible]	[illegible]	[illegible]	[illegible]	**************
[illegible]	[illegible]	2	0.00	[illegible]	[illegible]	[illegible]	[illegible]	4.246154

1058	173	2	0.00	9492.75	0.4750994E+03	1.0000	0.00	475.099400
1059	173	2	0.00	980760.90	0.2321544E+04	1.0000	0.00	2321.544000
1060	192	2	0.00	326.09	-0.1827529E+08	1.0000	0.00	**************
1061	192	2	0.00	9420.29	0.4246154E+01	1.0000	0.00	4.246154
1062	192	2	0.00	9492.75	0.4750994E+03	1.0000	0.00	475.099400
1063	192	2	0.00	980760.90	0.2321544E+04	1.0000	0.00	2321.544000
1064	211	2	0.00	326.09	-0.1827529E+08	1.0000	0.00	**************
1065	211	2	0.00	9420.29	0.4246154E+01	1.0000	0.00	4.246154
1066	211	2	0.00	9492.75	0.4750994E+03	1.0000	0.00	475.099400
1067	211	2	0.00	980760.90	0.2321544E+04	1.0000	0.00	2321.544000
1068	230	2	0.00	326.09	-0.1827529E+08	1.0000	0.00	**************
1069	230	2	0.00	9420.29	0.4246154E+01	1.0000	0.00	4.246154
1070	230	2	0.00	9492.75	0.4750994E+03	1.0000	0.00	475.099400
1071	230	2	0.00	980760.90	0.2321544E+04	1.0000	0.00	2321.544000

{Print Network matrix after solution}

```
Parameters associated with the Network Solver
_______________________________________________________________________

  SOURCE     SINK     NARC   NODES     IFS    IROOT     NDEG     NLOP     ITER   NPRIT
       1        2     1071     230       1        2       37      261     1041       0

  IPRINT
       0

         OUTFLO          FLONET          CSTNOW          TOTCST           EPS
   126749.00000    126749.00000         0.00000****************         0.00001

            BIG            SICH           TIMAX            TIME
  1000000.00000       223.48170         0.00000         0.00001

         OUTFLO          FLONET          CSTNOW          TOTCST           EPS
  0.1267490E+06   0.1267490E+06   0.0000000E+00  -0.1494077E+13   0.1000000E-04

            BIG            SICH           TIMAX            TIME
  0.1000000E+07   0.2234817E+03   0.0000000E+00   0.1000000E-04
```

Arc #	IARC	JARC	LOWER	UPPER	COST	AMP	FLOW	COST
1	1	2	0.00	126749.00	0.0000000E+00	1.0000	1252.45	0.000000
2	1	3	13900.00	13900.00	0.0000000E+00	1.0000	13900.00	0.000000
3	1	5	15605.00	15605.00	0.0000000E+00	1.0000	15605.00	0.000000
4	1	7	17739.00	17739.00	0.0000000E+00	1.0000	17739.00	0.000000
5	1	9	1730.00	1730.00	0.0000000E+00	1.0000	1730.00	0.000000
6	1	11	3473.00	3473.00	0.0000000E+00	1.0000	3473.00	0.000000
7	1	13	357.00	357.00	0.0000000E+00	1.0000	357.00	0.000000
8	1	3	722.00	722.00	0.0000000E+00	1.0000	722.00	0.000000
9	1	22	1197.00	1197.00	0.0000000E+00	1.0000	1197.00	0.000000
10	1	41	1429.00	1429.00	0.0000000E+00	1.0000	1429.00	0.000000
11	1	60	1790.00	1790.00	0.0000000E+00	1.0000	1790.00	0.000000
12	1	79	1323.00	1323.00	0.0000000E+00	1.0000	1323.00	0.000000
13	1	98	667.00	667.00	0.0000000E+00	1.0000	667.00	0.000000
14	1	117	714.00	714.00	0.0000000E+00	1.0000	714.00	0.000000
15	1	136	1064.00	1064.00	0.0000000E+00	1.0000	1064.00	0.000000
16	1	155	819.00	819.00	0.0000000E+00	1.0000	819.00	0.000000
17	1	174	497.00	497.00	0.0000000E+00	1.0000	497.00	0.000000
18	1	193	465.00	465.00	0.0000000E+00	1.0000	465.00	0.000000
19	1	212	445.00	445.00	0.0000000E+00	1.0000	445.00	0.000000
20	3	22	4211.00	10782.61	-0.2051292E+03	1.0000	10440.14	-205.129200
21	3	22	0.00	4289.86	-0.1234102E+03	1.0000	0.00	-123.410200
22	3	22	0.00	1594.20	-0.1844921E+02	1.0000	0.00	-18.449210
23	3	22	0.00	1047.33	0.1383689E+03	1.0000	0.00	138.368900
24	22	41	4211.00	11130.43	-0.2382582E+03	1.0000	10790.18	-238.258200
25	22	41	0.00	3884.06	-0.1191396E+03	1.0000	0.00	-119.139600
26	22	41	0.00	1565.22	0.5010889E+01	1.0000	0.00	5.010889
27	22	41	0.00	1134.29	0.1448219E+03	1.0000	0.00	144.821900
28	41	60	4211.00	11130.43	-0.2382582E+03	0.9983	11130.43	-238.258200
29	41	60	0.00	3884.06	-0.1191396E+03	0.9983	241.80	-119.139600
30	41	60	0.00	1565.22	0.5010889E+01	0.9983	0.00	5.010889
31	41	60	0.00	1134.29	0.1448219E+03	0.9983	0.00	144.821900
32	60	79	4211.00	11130.43	-0.2382582E+03	0.9988	11130.43	-238.258200
33	60	79	0.00	3884.06	-0.1191396E+03	0.9988	1165.40	-119.139600
34	60	79	0.00	1565.22	0.5010889E+01	0.9988	0.00	5.010889
35	60	79	0.00	1134.29	0.1448219E+03	0.9988	0.00	144.821900
36	79	98	4211.00	11072.46	-0.3223391E+03	0.9954	11072.46	-322.339100
37	79	98	0.00	3826.09	-0.1358066E+03	0.9954	1684.40	-135.806600
38	79	98	0.00	1652.17	0.1780186E+02	0.9954	0.00	17.801860
39	79	98	0.00	1163.28	0.2009804E+03	0.9954	0.00	200.980400
40	98	117	4211.00	11130.43	-0.2382582E+03	0.9951	11130.43	-238.258200
41	98	117	0.00	3884.06	-0.1191396E+03	0.9951	1387.28	-119.139600
42	98	117	0.00	1565.22	0.5010889E+01	0.9951	0.00	5.010889
43	98	117	0.00	1134.29	0.1448219E+03	0.9951	0.00	144.821900
44	117	136	4211.00	11043.48	-0.2218894E+03	0.9954	11043.48	-221.889400
45	117	136	0.00	3913.04	-0.1139978E+03	0.9954	845.75	-113.997800
46	117	136	0.00	1594.20	-0.8609623E+01	0.9954	0.00	-8.609623
47	117	136	0.00	1163.28	0.1333886E+03	0.9954	0.00	133.388600
48	136	155	4211.00	11130.43	-0.2382582E+03	0.9968	10323.53	-238.258200
49	136	155	0.00	3884.06	-0.1191396E+03	0.9968	0.00	-119.139600
50	136	155	0.00	1565.22	0.5010889E+01	0.9968	0.00	5.010889
51	136	155	0.00	1134.29	0.1448219E+03	0.9968	0.00	144.821900

{additional output deleted}

1021	229	230	0.00	9420.29	0.4246154E+01	1.0000	5929.29	4.246154
1022	229	230	0.00	9492.75	0.4750994E+03	1.0000	0.00	475.099400
1023	229	230	0.00	980760.90	0.2321544E+04	1.0000	0.00	2321.544000
1024	21	2	0.00	326.09	-0.1827529E+08	1.0000	326.09	**************
1025	21	2	0.00	9420.29	0.4246154E+01	1.0000	8271.38	4.246154
1026	21	2	0.00	9492.75	0.4750994E+03	1.0000	0.00	475.099400
1027	21	2	0.00	980760.90	0.2321544E+04	1.0000	0.00	2321.544000

{Last arcs in network matrix}

Arc #	IARC	JARC	LOWER	UPPER	COST	AMP	FLOW	COST
1028	40	2	0.00	326.09	-0.1827529E+08	1.0000	326.09	**************
1029	40	2	0.00	9420.29	0.4246154E+01	1.0000	8956.65	4.246154
1030	40	2	0.00	9492.75	0.4750994E+03	1.0000	0.00	475.099400
1031	40	2	0.00	980760.90	0.2321544E+04	1.0000	0.00	2321.544000
1032	59	2	0.00	326.09	-0.1827529E+08	1.0000	326.09	**************
1033	59	2	0.00	9420.29	0.4246154E+01	1.0000	5028.65	4.246154
1034	59	2	0.00	9492.75	0.4750994E+03	1.0000	0.00	475.099400
1035	59	2	0.00	980760.90	0.2321544E+04	1.0000	0.00	2321.544000
1036	78	2	0.00	326.09	-0.1827529E+08	1.0000	326.09	**************
1037	78	2	0.00	9420.29	0.4246154E+01	1.0000	7219.90	4.246154
1038	78	2	0.00	9492.75	0.4750994E+03	1.0000	0.00	475.099400
1039	78	2	0.00	980760.90	0.2321544E+04	1.0000	0.00	2321.544000
1040	97	2	0.00	326.09	-0.1827529E+08	1.0000	326.09	**************
1041	97	2	0.00	9420.29	0.4246154E+01	1.0000	8266.91	4.246154
1042	97	2	0.00	9492.75	0.4750994E+03	1.0000	0.00	475.099400
1043	97	2	0.00	980760.90	0.2321544E+04	1.0000	0.00	2321.544000
1044	116	2	0.00	326.09	-0.1827529E+08	1.0000	326.09	**************
1045	116	2	0.00	9420.29	0.4246154E+01	1.0000	3998.74	4.246154
1046	116	2	0.00	9492.75	0.4750994E+03	1.0000	0.00	475.099400
1047	116	2	0.00	980760.90	0.2321544E+04	1.0000	0.00	2321.544000
1048	135	2	0.00	326.09	-0.1827529E+08	1.0000	326.09	**************
1049	135	2	0.00	9420.29	0.4246154E+01	1.0000	9057.97	4.246154
1050	135	2	0.00	9492.75	0.4750994E+03	1.0000	0.00	475.099400
1051	135	2	0.00	980760.90	0.2321544E+04	1.0000	0.00	2321.544000
1052	154	2	0.00	326.09	-0.1827529E+08	1.0000	326.09	**************
1053	154	2	0.00	9420.29	0.4246154E+01	1.0000	6582.27	4.246154
1054	154	2	0.00	9492.75	0.4750994E+03	1.0000	0.00	475.099400
1055	154	2	0.00	980760.90	0.2321544E+04	1.0000	0.00	2321.544000
1056	173	2	0.00	326.09	-0.1827529E+08	1.0000	326.09	**************
1057	173	2	0.00	9420.29	0.4246154E+01	1.0000	5015.61	4.246154
1058	173	2	0.00	9492.75	0.4750994E+03	1.0000	0.00	475.099400
1059	173	2	0.00	980760.90	0.2321544E+04	1.0000	0.00	2321.544000
1060	192	2	0.00	326.09	-0.1827529E+08	1.0000	326.09	**************
1061	192	2	0.00	9420.29	0.4246154E+01	1.0000	4033.69	4.246154
1062	192	2	0.00	9492.75	0.4750994E+03	1.0000	0.00	475.099400
1063	192	2	0.00	980760.90	0.2321544E+04	1.0000	0.00	2321.544000
1064	211	2	0.00	326.09	-0.1827529E+08	1.0000	326.09	**************
1065	211	2	0.00	9420.29	0.4246154E+01	1.0000	5286.05	4.246154
1066	211	2	0.00	9492.75	0.4750994E+03	1.0000	0.00	475.099400
1067	211	2	0.00	980760.90	0.2321544E+04	1.0000	0.00	2321.544000

{The following 4 arcs are for the link between St. Louis and the Super Sink in February 1966}

1068	230	2	0.00	326.09	-0.1827529E+08	1.0000	326.09	**************
1069	230	2	0.00	9420.29	0.4246154E+01	1.0000	5929.29	4.246154
1070	230	2	0.00	9492.75	0.4750994E+03	1.0000	0.00	475.099400
1071	230	2	0.00	980760.90	0.2321544E+04	1.0000	0.00	2321.544000

{Check for continuity at nodes}

Check of feasibility

Arc #	Case	Flow	Parameter
323	Lower	0.000	0.000

Check for Optimality

Case	JUARC	ii	V	U/L Bound	COST	FLOW	AMP	POTi
Lower	50	155	0.863442E+03	0.00000	0.501089E+01	0.000	0.9968254	-0.240529E+03
Lower	57	193	0.452313E+03	0.00000	-0.101991E+03	0.000	0.9987790	-0.891309E+02
Lower	153	157	0.868192E+03	0.00000	0.156755E+02	0.000	0.9969749	-0.252303E+03
Upper	183	157	0.868192E+03	998176.80000	-0.235292E+02	0.000	1.0000000	-0.120483E+02
Upper	185	176	0.643435E+03	998176.80000	-0.235292E+02	0.000	1.0000000	-0.242734E+02
Upper	280	197	0.428813E+03	996973.90000	0.527123E+02	0.000	1.0000000	-0.594581E+03
Lower	317	66	-0.445168E+03	0.00000	0.319982E+03	0.000	0.9972952	-0.476706E+03
Lower	340	218	0.267063E+03	435.13050	-0.659559E+02	463.768	1.0000000	-0.743088E+03
Lower	438	163	-0.470688E+03	0.00000	0.994878E+03	0.000	0.9974481	-0.192237E+03
Lower	441	182	-0.390112E+03	0.00000	0.195578E+03	0.000	0.9976444	-0.115003E+03
Lower	442	182	-0.390112E+03	0.00000	0.918600E+03	0.000	0.9976444	-0.838025E+03
Lower	448	220	-0.819938E+03	0.00000	-0.800416E+03	1313.043	1.0000000	-0.420140E+03

Check for continuity at nodes

JNODES	V(JNODES)

{Write results to the output DSS data file}

```
-----DSS---ZWRITE Unit  73; Vers.    5:  //FTPK/STOR/01JAN1960/1MON/EXMPL1/
-----DSS---ZWRITE Unit  73; Vers.    5:  //GARR/STOR/01JAN1960/1MON/EXMPL1/
-----DSS---ZWRITE Unit  73; Vers.    5:  //OAHE/STOR/01JAN1960/1MON/EXMPL1/
-----DSS---ZWRITE Unit  73; Vers.    5:  //BEND/STOR/01JAN1960/1MON/EXMPL1/
-----DSS---ZWRITE Unit  73; Vers.    5:  //FTRA/STOR/01JAN1960/1MON/EXMPL1/
-----DSS---ZWRITE Unit  73; Vers.    5:  //GAPT/STOR/01JAN1960/1MON/EXMPL1/
-----DSS---ZWRITE Unit  73; Vers.    3:  //FTPK/FLOW_LOC/01JAN1960/1MON/EXMPL1/
-----DSS---ZWRITE Unit  73; Vers.    6:  //FTPK/STOR/01JAN1960/1MON/EXMPL1/
-----DSS---ZWRITE Unit  73; Vers.    3:  //FTPK/EVAP/01JAN1960/1MON/EXMPL1/
-----DSS---ZWRITE Unit  73; Vers.    3:  //FTPK-FTPKR/FLOW/01JAN1960/1MON/EXMPL1/
-----DSS---ZWRITE Unit  73; Vers.    3:  //FTPKR-GARR/FLOW/01JAN1960/1MON/EXMPL1/
-----DSS---ZWRITE Unit  73; Vers.    3:  //GARR/FLOW_LOC/01JAN1960/1MON/EXMPL1/
-----DSS---ZWRITE Unit  73; Vers.    6:  //GARR/STOR/01JAN1960/1MON/EXMPL1/
-----DSS---ZWRITE Unit  73; Vers.    3:  //GARR/EVAP/01JAN1960/1MON/EXMPL1/
-----DSS---ZWRITE Unit  73; Vers.    3:  //GARR-GARRR/FLOW/01JAN1960/1MON/EXMPL1/
-----DSS---ZWRITE Unit  73; Vers.    3:  //GARRR-OAHE/FLOW/01JAN1960/1MON/EXMPL1/
-----DSS---ZWRITE Unit  73; Vers.    3:  //OAHE/FLOW_LOC/01JAN1960/1MON/EXMPL1/
-----DSS---ZWRITE Unit  73; Vers.    6:  //OAHE/STOR/01JAN1960/1MON/EXMPL1/
-----DSS---ZWRITE Unit  73; Vers.    3:  //OAHE/EVAP/01JAN1960/1MON/EXMPL1/
-----DSS---ZWRITE Unit  73; Vers.    3:  //OAHE-OAHER/FLOW/01JAN1960/1MON/EXMPL1/
-----DSS---ZWRITE Unit  73; Vers.    3:  //OAHER-BEND/FLOW/01JAN1960/1MON/EXMPL1/
-----DSS---ZWRITE Unit  73; Vers.    6:  //BEND/STOR/01JAN1960/1MON/EXMPL1/
-----DSS---ZWRITE Unit  73; Vers.    3:  //BEND/EVAP/01JAN1960/1MON/EXMPL1/
-----DSS---ZWRITE Unit  73; Vers.    3:  //BEND-BENDR/FLOW/01JAN1960/1MON/EXMPL1/
-----DSS---ZWRITE Unit  73; Vers.    3:  //BENDR-FTRA/FLOW/01JAN1960/1MON/EXMPL1/
-----DSS---ZWRITE Unit  73; Vers.    3:  //FTRA/FLOW_LOC/01JAN1960/1MON/EXMPL1/
-----DSS---ZWRITE Unit  73; Vers.    6:  //FTRA/STOR/01JAN1960/1MON/EXMPL1/
-----DSS---ZWRITE Unit  73; Vers.    3:  //FTRA/EVAP/01JAN1960/1MON/EXMPL1/
-----DSS---ZWRITE Unit  73; Vers.    3:  //FTRA-FTRAR/FLOW/01JAN1960/1MON/EXMPL1/
-----DSS---ZWRITE Unit  73; Vers.    3:  //FTRAR-GAPT/FLOW/01JAN1960/1MON/EXMPL1/
-----DSS---ZWRITE Unit  73; Vers.    3:  //GAPT/FLOW_LOC/01JAN1960/1MON/EXMPL1/
-----DSS---ZWRITE Unit  73; Vers.    6:  //GAPT/STOR/01JAN1960/1MON/EXMPL1/
-----DSS---ZWRITE Unit  73; Vers.    3:  //GAPT/EVAP/01JAN1960/1MON/EXMPL1/
-----DSS---ZWRITE Unit  73; Vers.    3:  //GAPT-GAPTR/FLOW/01JAN1960/1MON/EXMPL1/
-----DSS---ZWRITE Unit  73; Vers.    3:  //GAPTR-SUX/FLOW/01JAN1960/1MON/EXMPL1/
-----DSS---ZWRITE Unit  73; Vers.    3:  //SUX/FLOW_LOC/01JAN1960/1MON/EXMPL1/
-----DSS---ZWRITE Unit  73; Vers.    3:  //SUX-OMA/FLOW/01JAN1960/1MON/EXMPL1/
-----DSS---ZWRITE Unit  73; Vers.    3:  //OMA/FLOW_LOC/01JAN1960/1MON/EXMPL1/
-----DSS---ZWRITE Unit  73; Vers.    3:  //OMA-NCNE/FLOW/01JAN1960/1MON/EXMPL1/
-----DSS---ZWRITE Unit  73; Vers.    3:  //NCNE/FLOW_LOC/01JAN1960/1MON/EXMPL1/
-----DSS---ZWRITE Unit  73; Vers.    3:  //NCNE-MKC/FLOW/01JAN1960/1MON/EXMPL1/
-----DSS---ZWRITE Unit  73; Vers.    3:  //MKC/FLOW_LOC/01JAN1960/1MON/EXMPL1/
-----DSS---ZWRITE Unit  73; Vers.    3:  //MKC-BNMO/FLOW/01JAN1960/1MON/EXMPL1/
-----DSS---ZWRITE Unit  73; Vers.    3:  //BNMO/FLOW_LOC/01JAN1960/1MON/EXMPL1/
-----DSS---ZWRITE Unit  73; Vers.    3:  //BNMO-HEMO/FLOW/01JAN1960/1MON/EXMPL1/
-----DSS---ZWRITE Unit  73; Vers.    3:  //HEMO/FLOW_LOC/01JAN1960/1MON/EXMPL1/
-----DSS---ZWRITE Unit  73; Vers.    3:  //HEMO-STL/FLOW/01JAN1960/1MON/EXMPL1/
-----DSS---ZWRITE Unit  73; Vers.    3:  //STL-S_SINK/FLOW/01JAN1960/1MON/EXMPL1/
```

{Close DSS data files and terminate}

```
    -----DSS---ZCLOSE Unit:  71,   File: TSIN.DSS
               Pointer Utilization:  0.40
               Number of Records:    453
               File Size:    451.3  Kbytes
               Percent Inactive:   0.0
    -----DSS---ZCLOSE Unit:  72,   File: PENCMP.DSS
               Pointer Utilization:  0.31
               Number of Records:    289
               File Size:    387.1  Kbytes
               Percent Inactive:   2.9
    -----DSS---ZCLOSE Unit:  73,   File: EXMPL1.DSS
               Pointer Utilization:  0.37
               Number of Records:    342
               File Size:    231.8  Kbytes
               Percent Inactive:   0.0

Elapsed CPU time is        66 seconds or       1.100 minutes or  0.018 hours.
```

Appendix G

HEC-PRM Pathname Parts (MRD)

HEC-PRM Pathname Parts

MRD Application

December 1991

The Hydrologic Engineering Center
Water Resources Support Center
U.S. Army Corps of Engineers
609 Second Street
Davis, California 95616-4687

(916) 756-1104

Table of Contents

List of Figures

List of Tables

Summary of Pathname Parts for HEC-PRM

Introduction

This text and the attached table summarizes pathnames as needed for HEC-PRM studies particularly as used in the MRD Study. The text written in capital letters must be defined exactly as written. Parts shown in lower case letters indicate that the user must define an appropriate part (e.g. "//b/..." might be replaced by "//FTPK/..."). The following characters have special meanings when used alone (separated by "/" or "-"):

"a" means part a of the pathname. Note that for Phase II, part "a" is eliminated for all node / link locations.

"b" means the node identifier for part "b" of the pathname.

"b1" means the node identifier for the upstream ("from") node of a link. It is used as a portion of pathname part "b".

"b2" means the node identifier for the downstream ("to") node of a link. It is used as a portion of pathname part "b".

"@" means the alternative identifier as entered on the "ZW" record in the HEC-PRM ASCII input data file.

Note that for time series data, part D is not shown as part of the pathname but it is assumed to be the standard military date required for regular interval time series data.

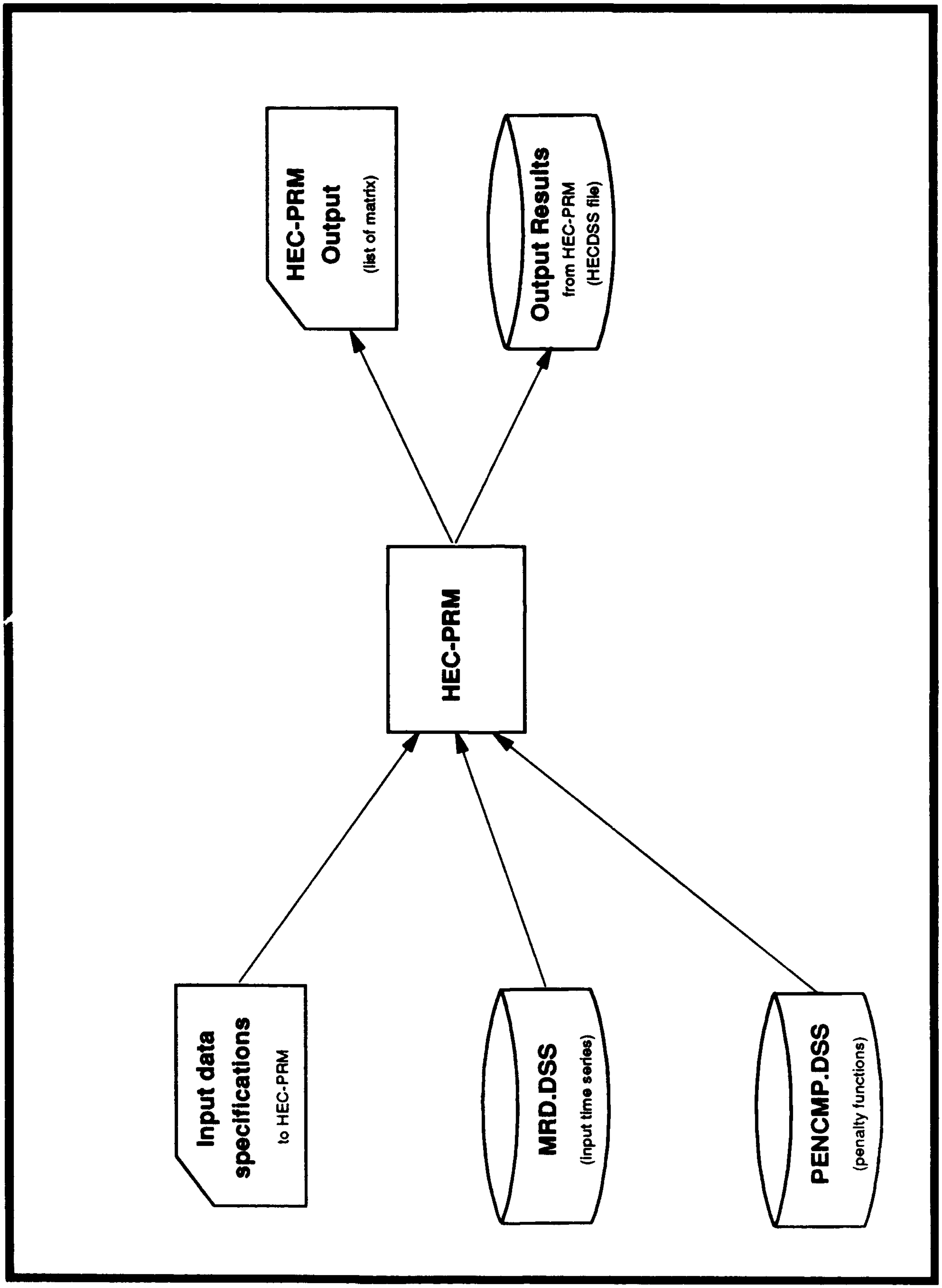
Maintaining Consistent Data Units

Since the same penalty function can be expressed in different units (e.g. thousands or millions of dollars), it is important to maintain some procedure or naming convention which allows you to accurately identify the data. The HEC-PRM model requires that storage and flow be expressed in the same units. It is easiest to conceptualize the data when it is expressed in 1,000 acre-feet per month as opposed to cubic-feet-per-second. However, the initial penalty functions might be derived in other units such as reservoir pool in terms of elevation. The table shows pathname part C to contain the units of the data for some types. However, it is recommended that you use the following procedure:

- Adopt consistent units for the flow/storage parameter (such as 1,000 acre-feet or 1,000 cubic meters) and for the penalty parameter (thousands of dollars).
- Enter the penalty functions and time series data in DSS data files. Do not include the units as part of the pathname part "c".
- If you enter the data in any other units besides the "standard, consistent" units you have adopted, then enter the data in one DSS data file and store the converted data in another file. This avoids the nuisance of entering data units in part C.

For example, if you adopt the storage units to be thousands of acre-feet (KAF) but you enter storage as elevation, then store the elevation-penalty functions in one DSS file (e.g. pfbase.dss) and use MATHPK to convert elevation to storage and write the storage-penalty function in another DSS file (e.g. pfmaster.dss) which is used as input to HEC-PRM.

Figure 27 HEC-PRM Data Relationships

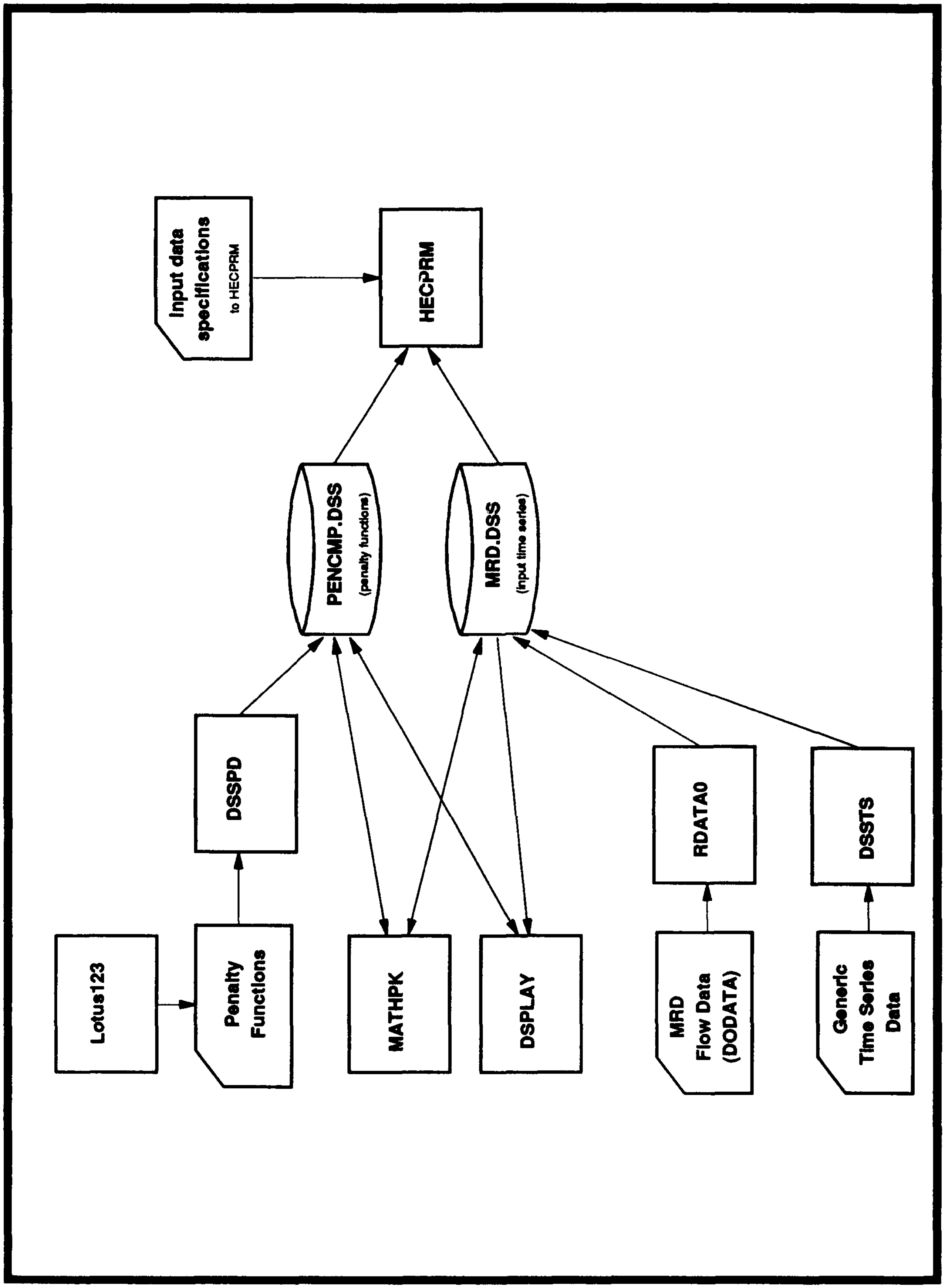


Summary of Developing Data With Recommended Pathname Parts

This section describes the process of entering data into DSS data files. The data may be entered by either running generalized HEC-DSS utility programs and typing data into the file from the keyboard or by executing District specific utility programs that read existing data from ASCII data files. Results are stored by one of several computer programs including HEC-PRM, PRMPOST, or by utility programs such as MATHPK. Some pathname parts are hardwired into HEC-PRM whereas others are dependent upon user input (node identifiers on "NODE" records and alternative identifiers on the "ZW" record). MATHPK uses parts defined by the user in either input commands or PREAD macros (file MPK.MAC). Specifically, the following programs may be used:

Program	Description
DSSPD	HEC-DSS generalized program which allows the user to store paired data in a DSS data file.
DSSTS	HEC-DSS generalized program which allows the user to store regular interval time series data in a DSS data file.
DSSUTL	HEC-DSS generalized program which allows the user to edit data which is already entered in a DSS data file.
DSPLAY	HEC-DSS generalized program which allows the user to graphically edit data which is already entered in a DSS data file. Graphical editing is done with a mouse.
MATHPK	HEC-DSS generalized program which allows the user to retrieve data from DSS data file(s), perform mathematical operation or functions on that data, and store the results in DSS data file(s).
RDATA0	MRD specific program which reads the file "D0DATA.D0D" which is a copy of MRD's file "D0DATA". RDATA0 provides default pathname parts and stores inflow, depletions, and evaporation rates in a DSS data file.
RDMATF	MRD specific program which reads "DMATFILE" files created by using MRD's program "V1.EXE". RDMATF defines default pathname parts and stores data associated with any of the parameters shown in the "V1.EXE" menu.
HEC-PRM	The optimization program saves flow, storage, and evaporation volume in an output DSS data file.

Figure 28 HEC-PRM Input Data Development



Enter and Compute Incremental Inflow, Depletions, and Adjusted Inflow Data

The unadjusted, incremental, local inflow is entered first. It is entered in the input DSS data file MRD.DSS. The basic flow and evaporation rate data is entered using "RDATA0". The monthly varying depletions are entered using the DSSPD program, and the adjusted local flows are computed and stored using MATHPK. All flow data is stored as regular interval time series data except the monthly varying depletions. They are stored as paired data - the first variable is the month number (1 through 12) and the second variable is the flow depletion. The following pathname parts are used:

Part	Description
A	Leave Blank.
B	The node identifier at which the inflow enters (e.g. "FTPK").
C	The parameter name including the code to indicate if the flow in total or incremental. For example:

Name	Description
FLOW_LOC_INC	Incremental local inflow.
FLOW_DPL	Incremental depletions from local inflow.
MON-FLOW_DPL	Monthly varying, incremental depletions from local inflow. (All January depletions are the same, all February depletions are the same but different than January).
FLOW_LOC	Incremental adjusted local inflow (computed from inflows and depletions).
FLOW_LOC_TOT	Total cumulative local inflow for all drainage above this location.
FLOW_DEPL_TOT	Total cumulative depletions.
FLOW_LOC_TOT	Total adjusted local inflow (computed).

Part	Description
D & E	Standard pathname parts for regular interval time series data.
F	Used only to identify the base year associated with depletions (e.g. 1975 level of depletion). The MATHPK macro which computes adjusted local inflow utilizes this

identifier to retrieve depletions for the desired year.

The following are example uses of pathname parts for the MRD study:

Local incremental inflow:
//FTPK/FLOW_LOC_INC/01JAN1900/1MON//
//SUX/FLOW_LOC_INC/01JAN1900/1MON//

Incremental depletions for 1975:
//FTPK/FLOW_LOC_DPL/01JAN1900/1MON/1975/
//SUX/FLOW_LOC_DPL/01JAN1900/1MON/1975/

Monthly varying, local incremental depletions (paired data):
//FTPK/MON-FLOW_DPL////
//SUX/MON-FLOW_DPL////

Incremental adjusted local inflow using depletions:
//FTPK/FLOW_LOC/01JAN1900/1MON//
//SUX/FLOW_LOC/01JAN1900/1MON//

The "Incremental adjusted local inflow using depletions" is used by HEC-PRM as basic input to define inflow to the system. Note that the depletion year is not used in that pathname.

Enter the Evaporation Rate

Currently, the yearly evaporation rate is read, distributed by month, and stored in the time series input DSS data file (MRD.DSS) by the program "RDATA0". "RDATA0" also stores the yearly rate. The distribution pattern is hardwired internally to RDATA0. You may develop a MATHPK macro to distribute it differently. The evaporation rates are stored as regular interval time series data. The following pathname parts are used:

Part	Description
A	Leave Blank.
B	The node identifier of the reservoir at which evaporation exits the network (e.g. "FTPK").
C	The parameter name including the code to indicate that it is an evaporation rate. It is assumed that the rate is for a time period which corresponds to the time interval. For example:

Name	Description
EVAP_RATE	Evaporation rate (in feet per month if entered as monthly regular interval data).
EVAP_RATE	Evaporation rate (in feet per year if entered as yearly regular interval data).

Part	Description
D & E	Standard pathname parts for regular interval time series data.
F	Leave it blank. It could be used to identify the base year (or method) associated with the evaporation. However, MATHPK macros currently require this to be left blank.

The following are example uses of pathname parts for the MRD study:

Monthly evaporation rate:
//FTPK/EVAP_RATE/01JAN1900/1MON//

Yearly evaporation rate:
//FTPK/EVAP_RATE/01JAN1900/1YEAR//

Monthly evaporation rate in inches per month. Normally, this should be stored in another "working" DSS file before conversion to the standard data units of feet per month:

//FTPK/EVAP_RATE(IM)/01JAN1900/1MON//

Enter the Elevation-Area-Capacity Curves

The elevation-area-capacity curves are used to convert reservoir storage to elevation or vice-versa or to plot the area-capacity curves. These curves are primarily used to post process results but can be used to convert penalty functions from elevation-penalty to storage-penalty. These curves are stored in the input DSS data file "PENCMP.DSS" using the generalized HEC-DSS program "DSSPD". The data is not exactly paired data because there are 3 parameters: elevation, area, and capacity. However, the data can be stored as paired data with elevation being the first variable and the second variable having 2 curves: (1) area and (2) capacity. To plot data with HECDSS-DSPLAY, the "TYpe" command must be used and the "CUrve" command may be used. The following pathname parts are used:

Part	Description
A	Leave Blank.
B	The node identifier of the reservoir for which the curves apply (e.g. "FTPK").
C	The parameter name "EL-AR-CAP".
D & E	Leave blank.
F	Leave blank. It could be used to identify the year in which the curve was calibrated. However, the MATHPK macros assume this part is blank.

The following are example uses of pathname parts for the MRD study:

Fort Peck rating curve for 1973:
//FTPK/EL-AR-CAP///1973/

Fort Peck rating curve used in the optimization model and MATHPK macros:
//FTPK/EL-AR-CAP////

Enter the Hydropower Coefficients

The hydropower coefficients are used to compute (by the MRD method) the amount of power and energy generated. The coefficients are stored in the input DSS data file PENCMP.DSS and are stored as paired data format. However, the data is not pure paired data since it consists of both coefficients and power data.

For all reservoirs except Gavins Point, the function contains three curves - the first variable is reservoir elevation (in FEET) and the second variable is power data. The first curve is the coefficient "A", the second curve is the coefficient "B", and the third curve is the capacity.
For Gavins Point, energy and power are a function of discharge and coefficients are not used. The first variable is discharge and the second variable is "POWER". The data is stored with three curves - the first curve is energy corresponding to the first variable which is discharge, the second curve is flow which corresponds to power, and the third curve is power which is related to the second curve.

The following pathname parts are used:

Part	Description
A	Leave Blank.
B	The node identifier of the reservoir for which the coefficients apply (e.g. "FTPK").
C	The parameter name "EL-COEFF_ENRGY" for all reservoirs except Gavins Point. "FLOW-POWER" for Gavins Point.
D & E	Leave blank.
F	Leave blank. It could be used to identify the method or year in which the curve was calibrated. However, MATHPK macros only recognize the pathname if part F is blank.

The following are example uses of pathname parts for the MRD study:

Fort Peck coefficients:
//FTPK/EL-COEFF_ENRGY////

Fort Peck coefficients in storage units:
//FTPK/STOR-COEFF_ENRGY////

Enter Class Intervals For Duration Analysis

The class intervals are used by MATHPK to compute the several functions relating reservoir elevation or flow and duration from HEC-PRM results. The functions include monthly as well as totals (e.g. the number of times in percent a given elevation was equalled or exceeded during all Januarys and for all months). The class intervals are stored in the input DSS data file PENCMP.DSS and are stored in paired data format. The first variable is the index number (1 through the number of values entered) and the second variable is the elevation (or flow) for each class interval. Elevations (or flow) are entered in increasing order (smallest values first). The DSSPD program may used to enter this data. However, if you define the class intervals in equal increments (e.g. elevation in 1 foot intervals), the MATHPK macro "CLASSDEF" may be used with much less effort. The following pathname parts are used:

Part	Description
A	Leave Blank.
B	The node identifier of the reservoir or the link for which the class intervals apply (e.g. "FTPK" or "FTPK-GARR").
C	The parameter name "INDEX-FLOW_CLASS_INTRVL" or "INDEX-ELEV_CLASS_INTRVL".
D & E	Leave blank.
F	Leave blank.

The following are example uses of pathname parts for the MRD study:

Fort Peck pool elevation class intervals:
//FTPK/INDEX-ELEV_CLASS_INTRVL////

Fort Peck release class intervals:
//FTPK-GARR/INDEX-FLOW_CLASS_INTRVL////

Enter the Storage Penalty Functions

The storage-penalty functions are entered as paired data convention in the input DSS data file PENCMP.DSS. Storage is entered in thousands of acre-feet (KAF) and penalty in thousands of dollars (K$). If the units are not the same as you have adopted, then the units must be included as part of part C or the functions must be stored in separate DSS data files. For all functions, the applying guidelines are suggested:

(1) Enter the basic twelve curves (one for each month of the year) in one pathname record. The first variable is "STOR" (storage in KAF) and the second is "PNLTY" (penalty in K$). Use the same values of storage for all penalty functions for one reservoir (e.g. recreation and water supply penalty functions are stored in separate pathnames but have the same storage ordinates). The parameter name "PNLTY" is appended with some qualifier label which identifies the penalty category (REC for recreation penalty due to storage etc.)

(2) If the function does not vary monthly (one curve applies to all months), enter the function as a monthly varying curve by duplicating the one curve for all months. Current MATHPK macros assume all functions vary monthly so the curve should be duplicated for all months before storing in the DSS data file.

(3) The computed composite penalty function is calculated using MATHPK to add the component parts (e.g. recreation, water supply, etc.) to get the total penalty for each storage. It is then stored in the DSS data file "PENCMP.DSS".

(4) Store the edited composite penalty functions as a single curve for a given location for a given month (e.g. the edited Fort Peck July penalty function is stored in one pathname and the August penalty function is stored in a different record). The edited composite function is typically derived using the graphical editing capability of HECDSS-DSPLAY. However, the data may be exported to some other package (such as Excel) for graphical editing.

(5) To mesh with MATHPK macros, the final, edited penalty functions should have a blank pathname part F.

The following pathname parts are used for storage penalty functions:

Part	Description
A	Leave Blank.
B	The node identifier of the reservoir for which the penalty function(s) apply (e.g. "FTPK").
C	The parameter names "STOR-PNLTY_###" where "###" is the qualifier for the penalty. The following are typical definitions for "###":

Code	Description
NAV	Navigation
WSP	Water Supply
HPC	Hydropower Capacity
HPE	Hydropower Energy
REC	Recreation
FDU	Flood Damage Urban.
FDA	Flood Damage Agricultural.
CMP	Computed composite curve.
EDT	Edited composite curve.

Part	Description
D & E	Leave blank.
F	Leave this part blank for the final functions including the **individual category functions** (e.g. recreation), the **computed composite, and the edited composite** penalty functions. It could be used to identify the version of this function such as "INITL" for the original function, "VER1" for version 1 of the composite computed function, or "VER2" for version 2 of the composite edited function. However, MATHPK macros require a blank part F.

The following are example uses of pathname parts for the MRD study:

Fort Peck pool recreation:
//FTPK/STOR-PNLTY_REC////

Fort Peck pool hydropower capacity:
//FTPK/STOR-PNLTY_HPC////

Fort Peck pool total composite computed:
//FTPK/STOR-PNLTY_CMP////

Fort Peck pool preliminary, total, composite, edited:
//FTPK/STOR-PNLTY_EDT///VER1/

Fort Peck pool final, total, composite, edited (used in the model):
//FTPK/STOR-PNLTY_EDT////

Enter the Flow Penalty Functions

The flow-penalty functions are entered as paired data convention in the input DSS data file PENCMP.DSS. Flow is entered in thousands of acre-feet per month (KAF) and penalty in thousands of dollars (K$). If the units are not the same as you have adopted, then the units must be included as part of part C or the functions must be stored in separate DSS data files. For all functions, the applying guidelines are suggested:

(1) Enter the **basic** penalty functions as twelve curves (one for each month of the year) in one pathname record. The first variable is "FLOW" (flow in KAF) and the second is "PNLTY" (penalty in K$). The parameter name "PNLTY" is appended by some qualifier label which identifies the penalty category (REC for recreation penalty due to flow, etc.)

(2) If the function does not vary monthly (one curve applies to all months), enter the function as a monthly varying curve by duplicating the one curve for all months. Current MATHPK macros assume all functions vary monthly so the curve should be duplicated for all months before storing in the DSS data file.

(3) Store the **hydropower energy** penalty functions as a family of curves in which each pathname record contains data for one location and one month of the year. Hydropower energy penalties are a function of both reservoir release and pool storage. Each curve has an associated pool storage in KAF. The pool storage is entered as the "curve label" in accordance with standard DSS conventions.

(4) The **computed composite** penalty function is calculated using MATHPK to add the component parts (e.g. recreation, water supply, etc.) to get the total penalty for each flow. Since there is a separate link for hydropower releases, the computed composite curve does not contain any energy penalties. It is then stored in the DSS data file "PENCMP.DSS".

(5) Store the **edited composite** penalty functions as a single curve for a given location for a given month (e.g. the edited Fort Peck to Garrison July penalty function is stored in one pathname and the August penalty function is stored in a different record). The **edited composite** function is typically derived using the graphical editing capability of HECDSS-DSPLAY. However, the data may be exported to some other package (such as Excel) for graphical editing.

(6) To mesh with MATHPK macros, the final, edited penalty functions should have a blank pathname part F.

The following pathname parts are used:

Part	Description
A	Leave Blank.
B	The identifiers of the upstream (or "from" node) and the downstream (or "to" node).
C	The parameter names "FLOW-PNLTY_###" where "###" is the qualifier for the penalty. The following are typical definitions for "###":

Code	Description
NAV	Navigation
WSP	Water Supply
HPC	Hydropower Capacity
HPE	Hydropower Energy
REC	Recreation
FDU	Flood Damage Urban.
FDA	Flood Damage Agricultural.
HPE_EDT	Edited hydropower energy curve.
CMP	Computed composite curve.
EDT	Edited composite curve.

Part	Description
F	Leave this part blank for the final functions including the **individual category functions** (e.g. recreation), the **computed composite**, and the **edited composite** penalty functions. It could be used to identify the version of this function such as "INITL" for the original function, "VER1" for version 1 of the composite computed function, or "VER2" for version 2 of the composite edited function. However, MATHPK macros require a blank part F.

The following are example uses of pathname parts for the MRD study:

Fort Peck release recreation:
//FTPK-GARR/FLOW-PNLTY_REC////

Fort Peck hydropower energy for March:
//FTPK-GARR/FLOW-PNLTY_HPE//MAR//

Fort Peck computed, composite penalty for releases:
//FTPK-GARR/FLOW-PNLTY_CMP////

Fort Peck edited, penalty for hydropower releases for July:
//FTPK-GARR/FLOW-PNLTY_HPE_EDT//JUL//

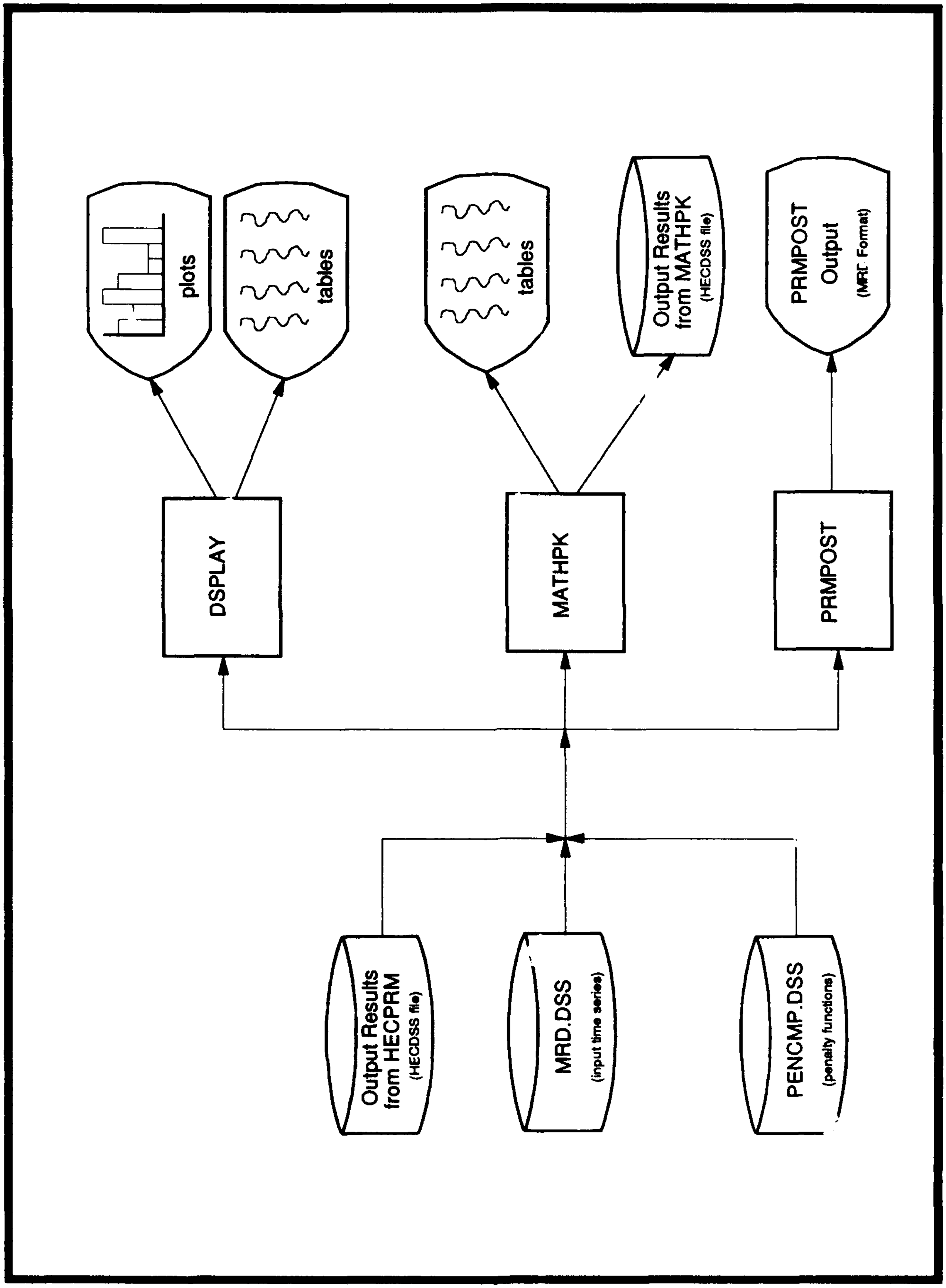
Note: The edited hydropower energy curves are a family of curves, each curve corresponds to an assumed pool storage. The same flows must be used for all curves.

Fort Peck preliminary, edited, composite penalty for releases in July:
//FTPK-GARR/FLOW-PNLTY_EDT//JUL/VER1/

Fort Peck final, edited, composite penalty for releases in July:
//FTPK-GARR/FLOW-PNLTY_EDT//JUL//

Sioux City navigation:
//SUX-OMA/FLOW-PNLTY_NAV////

Figure 29 HEC-PRM Post-Processing



Writing Evaporation, Storage, and Discharge Results To DSS by HEC-PRM

HEC-PRM writes three basic "water" parameters (as opposed to "economic" parameters) to an output DSS data file: (1) Evaporation flow in KAF/month, (2) Reservoir storage in KAF, and (3) Flow in KAF/month (either channel, diversion, or release). Normally, there is one output DSS data file for each alternative. HEC-PRM utilizes input from the "ZW" record to define part F. The following pathname parts are used for regular interval time series evaporation, reservoir storage, and discharge:

Part	Description
A	Leave Blank.
B	For reservoir storage and evaporation, the node identifier of the reservoir (e.g. "FTPK"). For flows (channel, release, diversion) the upstream ("from") node and downstream ("to") node identifiers (e.g. "FTPK-GARR").
C	The parameter name. Storage is "STOR" and evaporation is "EVAP" and flow is "FLOW". To determine the type of flow (channel, diversion, or release), the user must consult HEC-PRM input.
D & E	Standard pathname parts for regular interval time series data.
F	Identifies the alternative being studied. It is taken from the ZW record in the HEC-PRM ASCII input data file.

The following are example uses of pathname parts for the MRD study where the alternative identifier from the ZW record is "5Y0D":

Reservoir storage in Fort Peck:
//FTPK/STOR/01JAN1960/1MON/5Y0D/

Reservoir evaporation from Fort Peck:
//FTPK/EVAP/01JAN1960/1MON/5Y0D/

Reservoir release from Fort Peck (hydropower and "all other purposes" links):
//FTPK-FTPKR/FLOW/01JAN1960/1MON/5Y0D/
//FTPKR-GARR/FLOW/01JAN1960/1MON/5Y0D/

Channel flow from Sioux City to Omaha:
//SUX-OMA/FLOW/01JAN1960/1MON/5Y0D/

Hypothetical diversion from Gavins Point to Omaha:
//GAPT-OMA/FLOW/01JAN1960/1MON/5Y0D/

Writing Edited Penalty Functions To DSS by HEC-PRM

HEC-PRM writes edited penalty functions to the **output** DSS data file. Since HEC-PRM input allows one edited function to apply to more than one month, it computes functions for each month, extends or truncates the function at zero storage or flow, and extends or truncates the function at the upper bound as entered in HEC-PRM input. The following pathname parts are used for regular interval evaporation, reservoir storage, and discharge:

Part	Description
A	Leave Blank.
B	For reservoir storage, the node identifier of the reservoir (e.g. "FTPK"). For flows (channel, release, diversion) the upstream ("from") node and downstream ("to") node identifiers (e.g. "FTPK-GARR").
C	The parameter names such as "STOR-PNLTY_EDT" or "FLOW-PNLTY_EDT", or "FLOW-PNLTY_HPE_EDT".
D	Leave blank.
E	The 3 character identifier of the month for which the penalty function applies (e.g. "JUL").
F	Identifies the alternative being studied. It is taken from the ZW record in the HEC-PRM ASCII input data file.

The following are example uses of pathname parts for the MRD study where the alternative identifier from the ZW record is "5Y0D":

Reservoir storage penalty for Fort Peck for July:
//FTPK/STOR-PNLTY_EDT//JUL/5Y0D/

Reservoir release from Fort Peck (hydropower and "all other purposes" links) for July:
//FTPK-FTPKR/FLOW-PNLTY_HPE_EDT//JUL/5Y0D/
//FTPKR-GARR/FLOW-PNLTY_EDT//JUL/5Y0D/

Channel flow from Sioux City to Omaha for July:
//SUX-OMA/FLOW-PNLTY_EDT//JUL/5Y0D/

Hypothetical diversion from Gavins Point to Omaha:
//GAPT-OMA/FLOW-PNLTY_EDT//JUL/5Y0D/

Writing Converted Results To DSS by MATHPK

The three basic "water" parameters written to an output DSS data file are converted to other units by application of the MATHPK program and associated macros: (1) Evaporation flow in KAF/month is not converted, (2) Reservoir storage in KAF is converted to pool elevation in feet, and (3) Flow in KAF/month (either channel, diversion, or release) is converted to KCFS (thousands of cubic-feet-per-second). MATHPK opens another DSS file which has the same name as the HEC-PRM output DSS data file but appends the letter "O" to it (e.g. I5Y0D.DSS to I5Y0DO.DSS). MATHPK modifies the pathname of the basic results to store converted results. The following pathname parts are used for regular interval reservoir storage and discharge:

Part	Description
A	Left Blank.
B	For reservoir storage, the node identifier of the reservoir (e.g. "FTPK"). For flows (channel, release, etc.) the upstream ("from") node and downstream ("to") node identifiers (e.g. "FTPK-GARR").
C	The parameter name. Pool elevation is "ELEV", flow is "FLOW". Flow is qualified by the units when they don't conform to the standard (KAF) such as "FLOW(CFS)".
D & E	Standard pathname parts for regular interval time series data.
F	Identifies the alternative being studied. It is taken from the ZW record in the HEC-PRM ASCII input data file.

The following are example uses of pathname parts for the MRD study where the alternative identifier from the ZW record is "5Y0D":

Reservoir storage in Fort Peck:
//FTPK/STOR/01JAN1960/1MON/5Y0D/
//FTPK/ELEV/01JAN1960/1MON/5Y0D/

Reservoir release from Fort Peck:
//FTPK-GARR/FLOW/01JAN1960/1MON/5Y0D/
//FTPK-GARR/FLOW(CFS)/01JAN1960/1MON/5Y0D/

Channel flow from Sioux City to Omaha:
//SUX-OMA/FLOW/01JAN1960/1MON/5Y0D/
//SUX-OMA/FLOW(CFS)/01JAN1960/1MON/5Y0D/

Hypothetical diversion from Gavins Point to Omaha:
//GAPT-OMA/FLOW(CFS)/01JAN1960/1MON/I5Y0D/

Writing Economic (Penalty) Results To DSS by MATHPK

HEC-PRM writes basic "water" parameters to an output DSS data file: (1) Evaporation flow in KAF/month, (2) Reservoir storage in KAF, and (3) Flow in KAF/month (either channel, diversion, or release). MATHPK utilizes these results and the penalty functions to compute regular interval time series penalty histograms. MATHPK opens another DSS file which has the same name as the HEC-PRM output DSS data file but appends the letter "O" to it (e.g. I5Y0D.DSS to I5Y0DO.DSS). MATHPK modifies the pathname of the basic results to store converted results. The following pathname parts are used for penalty histograms based upon regular interval time series reservoir storage and discharge:

Part	Description
A	Leave Blank.
B	For reservoir storage, the node identifier of the reservoir (e.g. "FTPK"). For flows (channel, release, etc.) the upstream ("from") node and downstream ("to") node identifiers (e.g. "FTPK-GARR").
C	The penalty code name "PNLTY" appended by the type of category (e.g. "PNLTY_REC" for penalties associated with recreation. MATHPK macros compute the difference between penalties calculated with the "computed composite" penalty function and the "edited composite" penalty function. This time series is identified as "PNLTY_DIFF".
D & E	Standard pathname parts for regular interval time series data.
F	Identifies the alternative being studied. It is taken from the ZW record in the HEC-PRM ASCII input data file.

The following are example uses of pathname parts for the MRD study where the alternative identifier from the ZW record is "5Y0D"::

Recreation penalty of Fort Peck pool:
//FTPK/PNLTY_REC/01JAN1960/1MON/5Y0D/

Hydropower capacity penalty of Fort Peck pool:
//FTPK/PNLTY_HPC/01JAN1960/1MON/5Y0D/

Total penalty of Fort Peck pool using composite computed function:
//FTPK/PNLTY_CMP/01JAN1960/1MON/5Y0D/

Total penalty of Fort Peck pool using composite edited function (used in the HEC-PRM model:
//FTPK/PNLTY_EDT/01JAN1960/1MON/5Y0D/

Recreation penalty of Fort Peck release:
//FTPK-GARR/PNLTY_REC/01JAN1960/1MON/5Y0D/

Energy penalty of Fort Peck hydropower release using computed function:
//FTPK-GARR/PNLTY_HPE/01JAN1960/1MON/5Y0D/

Energy penalty of Fort Peck hydropower release using edited function:
//FTPK-GARR/PNLTY_HPE_EDT/01JAN1960/1MON/5Y0D/

Total penalty of Fort Peck release using composite computed penalty function:
//FTPK-GARR/PNLTY_CMP/01JAN1960/1MON/5Y0D/

Total penalty of Fort Peck release using the edited composite penalty function:
//FTPK-GARR/PNLTY_EDT/01JAN1960/1MON/5Y0D/

Navigation penalty of Sioux City flow:
//SUX-OMA/PNLTY_NAV/01JAN1960/1MON/5Y0D/

Writing Hydropower and Duration Results To DSS by MATHPK

MATHPK calculates and writes computed power and duration results to the same output DSS data file that the converted results (e.g. elevation or flow in cfs) and penalty histograms are written. For example, if HEC-PRM write results to I5Y0D.DSS, MATHPK reads that file, converts data (e.g. storage to elevation or time series penalties), and stores the results in the file I5Y0DO.DSS. MATHPK also computes hydropower and duration data and stores it to the file I5Y0DO.DSS.

MRD hydropower calculations utilize time series data, such as elevation and flow, as well as the coefficients "A" and "B" as defined in MRD's Long Range Planning Model Program Description. The also utilize tables of elevation versus power capacity. Elevations and flows are computed by HEC-PRM and MATHPK macros. The coefficients and tables are entered by the analyst using DSSPD and the pathname parts are described earlier. Duration results utilize a table of class intervals to calculate the number of occurrences within each class, the cumulative totals, and the percent exceedance for each interval. The tables of class intervals may be entered manually or calculated by MATHPK, but they must be stored in the DSS data file PENCMP.DSS.

Various MATHPK macros compute time series histograms of power and paired data functions of duration. The following pathname parts are used for hydropower regular interval time series results:

Part	Description
A	Leave Blank.
B	The node identifier of the reservoir (e.g. "FTPK").
C	The parameter identifier - "ENERGY" for energy generated and "POWER_PEAK" for peak power.
D & E	Standard pathname parts for regular interval time series data.
F	Identifies the alternative being studied. It is taken from the ZW record in the HEC-PRM ASCII input data file.

Various MATHPK macros compute duration functions and store them as paired data. The calculated functions include: (1) The number of occurrences within a given class interval, (2) The cumulative total occurrences starting from the smallest value, (3) The cumulative total occurrences starting from the highest value, (4) The number of values (in percent) which equal or exceed a given class interval. The following pathname parts are duration paired data functions:

Part	Description
A	Leave Blank.
B	For reservoir storage, the node identifier of the reservoir (e.g. "FTPK"). For flows (channel, release, etc.) the upstream ("from") node and downstream ("to") node identifiers (e.g. "FTPK-GARR").
C	The parameter identifier and type of duration calculation. The first variable is the parameter (e.g. "ELEV" for elevation) and the second variable identifies the type of duration function. The duration functions include (shown with the parameter "ELEV" for example): "ELEV-NU_INCLASS" contains the number of occurrences within a given class interval, "ELEV-NU_INCLASS_CUMLO" contains the cumulative number of occurrences within a given class starting with the lowest value, "ELEV-NU_INCLASS_CUMHI" contains the cumulative number of occurrences within a given class starting with the highest value, and "ELEV-NU_INCLASS_CUMHI_PCT" contains the number of values (in percent) which equal or exceed a given class interval.
D & E	Leave blank.
F	Identifies the alternative being studied. It is taken from the ZW record in the HEC-PRM ASCII input data file.

The following are example uses of pathname parts for the MRD study where the alternative identifier from the ZW record is "5Y0D"::

Peak power at Fort Peck:
//FTPK/POWER_PEAK/01JAN1960/1MON/5Y0D/

Energy generated at Fort Peck:
//FTPK/ENERGY/01JAN1960/1MON/5Y0D/

Duration analysis of Fort Peck Pool elevation:
//FTPK/ELEV-NU_INCLASS///5Y0D/
//FTPK/ELEV-NU_INCLASS_CUMLO///5Y0D/
//FTPK/ELEV-NU_INCLASS_CUMHI///5Y0D/
//FTPK/ELEV-NU_INCLASS_CUMHI_PCT///5Y0D/

Duration analysis of Fort Peck Release:
//FTPK-GARR/FLOW-NU_INCLASS///5Y0D/
//FTPK-GARR/FLOW-NU_INCLASS_CUMLO///5Y0D/
//FTPK-GARR/FLOW-NU_INCLASS_CUMHI///5Y0D/
//FTPK-GARR/FLOW-NU_INCLASS_CUMHI_PCT///5Y0D/

HEC-DSS Pathname Part Conventions for HEC-PRM

Time Series - Input

Parameter	Units	Example Path	Description
Inflow - Incremental (unadjusted)	KCFS KAF CMS	//b/FLOW_LOC_INC(KCFS)/01JAN1960/1MON// //b/FLOW_LOC_INC/01JAN1960/1MON//	Incremental local inflow. This is the raw inflow which has not been adjusted for depletions.
Inflow - Cumulative Total	KCFS KAF CMS	//b/FLOW_LOC_TOT(KCFS)//1MON// //b/FLOW_LOC_TOT//1MON//	Total inflow (cumulative from all upstream drainage area. This is not used in MRD study because all inflows are incremental.
Depletions	KCFS KAF KAF CMS	//b/FLOW_DPL//1MON// //b/MON-FLOW_DPL////	Incremental depletions to this reach. Depletions can be of 2 types: (1) regular interval time series, and (2) monthly varying stored as paired data.
Adjusted inflow	KCFS KAF CMS	//b/FLOW_LOC(KCFS)//1MON// //b/FLOW_LOC//1MON//	Adjusted local incremental inflow (add incremental local to depletions, both regular interval and monthly varying).

Evaporation rate.	IN/MONTH
	FT/YEAR
	FT/MONTH
		//b/EVAP_RATE(IM)//1MON//
		//b/EVAP_RATE//1YEAR//
		//b/EVAP_RATE//1MON//
			Evaporation rate.

Evaporation Distribution.	%	//b/MON-EVAP_RATE_DISTRIB////	Monthly distribution (in percent) of yearly evaporation rate over 12 months. Stored as paired data.
Month number.		///MON//1MON//	Integer number identifying month (1-12).
Year number.		///YR//1MON//	Integer number identifying the year (e.g. 1991).
Conversion factor.	CFS / KAF/MON	/FACTOR//CFS-KAF//1MON//	Factor to convert flow in CFS into KAF per month by multiplication. (Conversely, divide to convert KAF into CFS).
Season Identifier.		///NAV_SEASON////	Flag indicating navigation season. It is set to 1 for the months April through November and set to 0 (zero) for the months December through March.

Time Series - Output

Parameter	Units	Example Path	Description
Evaporation	KCFS KAF CMS	//b/EVAP(KAF)/01JAN1960/1MON/@/	Evaporation out of the reservoir.
Flow - Reservoir release	KCFS CFS KAF CMS	//b1-b2/FLOW(CFS)/01JAN1960/1MON/@/ //b1-b2/FLOW/01JAN1960/1MON/@/	Reservoir release. There are two types (links) associated with the reservoir release: (1) hydropower energy, and (2) all other purposes. The flow rates are the same as the links are in series with no intervening area local inflow. There is no difference in pathname parts between channel, diversion, and reservoir release links. The type of link is determined by the HEC-PRM input data set.
Flow - channel	KCFS CFS KAF CMS FT	//b1-b2/FLOW(CFS)/01JAN1960/1MON/@/ //b1-b2/FLOW/01JAN1960/1MON/@/	Channel flow. There is no difference in pathname parts between channel, diversion, and reservoir release links. The type of link is determined by the HEC-PRM input data set.
Flow - diversion	KCFS CFS KAF CMS FT	//b1-b2/FLOW(CFS)/01JAN1960/1MON/@/ //b1-b2/FLOW/01JAN1960/1MON/@/	Diversion flow from one node to another. There is no difference in pathname parts between channel, diversion, and reservoir release links. The type of link is determined by the HEC-PRM input data set.
Storage - Volume	KAF	//b/STOR/01JAN1960/1MON/@/	Reservoir storage (end of period).
Storage - Height	FT	//b/ELEV/01JAN1960/1MON/@/	Reservoir Pool elevation corresponding to reservoir storage (volume).

Penalty - Storage	M$
	K$
		//b/PNLTY_EDT(M$)/01JAN1960/1MON/@/
		//b/PNLTY_REC/01JAN1960/1MON/@/
		//b/PNLTY_WSP/01JAN1960/1MON/@/
		//b/PNLTY_HPC/01JAN1960/1MON/@/
			Penalty of storing water in reservoir pool. Pathname part B determines the type of penalty - either storage or flow. Part B contains one node identifier for storage and both node identifiers (from and to) for flow penalty functions. Part C includes the type of penalty (REC is recreation, EDT is convex edited function, etc.)

Penalty - Channel	M$
	K$
		//b1-b2/PNLTY_EDT(M$)/01JAN1960/1MON/@/
		//b1-b2/PNLTY_REC/01JAN1960/1MON/@/
		//b1-b2/PNLTY_FDA/01JAN1960/1MON/@/
		//b1-b2/PNLTY_HPE_EDT/01JAN1960/1MON/@/
			Penalty of releasing water through this channel. Pathname part B determines the type of penalty - either storage or flow. Part B contains one node identifier for storage and both node identifiers (from and to) for flow penalty functions. Part C includes the type of penalty (REC is recreation, EDT is convex edited function, etc.

Hydropower - Energy	KMWH	//b/ENERGY/01JAN1960/1MON/@/	Amount of energy generated for reservoir "b".
Hydropower - Capacity	MW	//b/POWER_PEAK/01JAN1960/1MON/@/	Peak hydropower capacity at reservoir "b".

Paired Data - Input

Parameter	Units	Example Path	Description
Elevation-Area-Capacity	FT-AC-AF FT-KAC-KAF	//b/EL-AR(AC)-CAP(AF)//// //b/EL-AR-CAP////	Elevation-area-capacity curve for a reservoir.
Hydropower - Energy Coefficients & Capacity Tables		//b/EL-COEFF_ENRGY//// //GAPT/FLOW-POWER////	First column contains elevation, second and third columns contain coefficients A and B, and the fourth column contains power capacity corresponding to elevations in column 1. Gavins Point is special case where energy and capacity are related only to flow (pool elevation is assumed constant).
Duration - Class Intervals		//b/INDEX-ELEV_CLASS_INTRVL//// //b1-b2/INDEX-FLOW_CLASS_INTRVL////	Class intervals used in duration analysis. First column is index number (1-number of points) and second column is parameter value for class interval in increasing order.
Penalty - Storage	KAF-K$	//b/STOR-PNLTY_CMP//// //b/STOR-PNLTY_REC//// //b/STOR-PNLTY_EDT//JAN//	Storage penalties. First column is storage in KAF, second column is penalties for January, third column is penalties for February, etc. for all 12 months. Edited function contains data for one month and month is entered as part E of pathname.
Penalty - Flow	KAF-K$	//b/FLOW-PNLTY_CMP//// //b/FLOW-PNLTY_REC//// //b/FLOW-PNLTY_EDT//JUL//	Flow penalties. First column is flow in KAF/month, second column is penalties for January, third column is penalties for February, etc. for all 12 months. Edited function contains data for one month and month is entered as part E of pathname.
Penalty - Hydropower Flow	KAF-K$	//b1-b2/FLOW-PNLTY_HPE//JUL// //b1-b2/FLOW-PNLTY_HPE_EDT//JUL//	Flow versus penalty as a function of reservoir storage (elevation). Each record contains several curves, each for a different assumed reservoir elevation. The edited functions are stored in a separate pathname for each month.
Depletions - Monthly varying	KAF	//b/MON-FLOW_DPL////	Incremental depletions to this reach. Depletions can be of 2 types: (1) regular interval time series, and (2) monthly varying stored as paired data. The monthly varying is stored as paired function data.

Paired Data - Output

Parameter	Units	Example Path	Description

Penalty - Edited Function	KAF-K$	//b/STOR-PNLTY_EDT//JUL/@/
		//b1-b2/FLOW-PNLTY_EDT//JUL/@/
		//b1-b2/FLOW-PNLTY_HPE_EDT//JUL/@/
			Edited penalty functions for reservoir "b" and link "b1" to "b2". HEC-PRM extends / truncates original penalty functions to include zero flow/storage and the upper bound. The function for each month is stored in separate pathnames (records).

Duration Analysis - Number of Occurrences		//b/ELEV-NU_INCLASS///@/ //b1-b2/FLOW-NU_INCLASS///@/	Number of occurrences within each class interval.
Duration Analysis - Cumulative occurrences		//b/ELEV-NU_INCLASS_CUMLO///@/ //b/ELEV-NU_INCLASS_CUMHI///@/	Cumulative number of occurrences in increasing order or decreasing order.
Duration Analysis - Percent Exceeding		//b/ELEV-NU_INCLASS_CUMHI_PCT///@/ //b1-b2/FLOW-NU_INCLASS_CUMHI_PCT///@/	Number of occurrences equaling or exceeding this class interval.

Appendix H

HEC-PRM Supplemental Programs (MRD)

HEC-PRM
Supplemental Programs

MRD Application

December 1991

The Hydrologic Engineering Center
Water Resources Support Center
U.S. Army Corps of Engineers
609 Second Street
Davis, California 95616-4687

(916) 756-1104

Table of Contents

List of Figures

List of Tables

RDATA0

Purpose of RDATA0

RDATA0 reads MRD's file "DATA0" and stores the data in a HECDSS data file. RDATA0 is tailored specifically to the format of this file --- all data is read in a "fixed format". It stores monthly inflow, depletions, and evaporation rates. It reads yearly evaporation rates from "DODATA" and distributes it amongst the months by an internally stored distribution. Any changes to this distribution must be made internally to the program.

Copying the Program onto Your Hard Disk

Use the "pkunzip.exe" utility to retrieve and uncompress the program from diskette. If "pkunzip.exe" does not already exist on your system, copy it from the diskette onto your fixed disk drive (hard disk). Normally, it would be copied into the "C:\HECEXE" subdirectory and the "path" command in your "C:\AUTOEXEC.BAT" file would contain a reference to this subdirectory. To retrieve the program from diskette in drive "a:" and store it on your fixed disk drive in subdirectory "c:\hecexe", enter the following command:

```
pkunzip a:rdata0 c:\hecexe rdata0.exe
```

Make sure you use the character "zero" as the last character in the name "RDATA0" (as opposed to the letter "oh").

Copying Example Data onto Your Hard Disk

To copy the example input data file "rdata0.in" onto your fixed disk drive (hard disk) from the floppy diskette drive "a:", enter the following command:

pkunzip a:rdata0in rdata0.in

This will store the example data in your default directory (the directory in which you were at the time of issuing the command).

Preparing the Input data file.

The user need not make any changes to the existing "DATA0" file --- "RDATA0" can read it without modifications. The following is an example section of the file "DATA0".

1898	1	2.30		1.79		1.61		1.12		1.51		1.17		EVAP
1898	11	419	776	1529	2786	1342	386	320	353	339	313	232	232	413
1898	12	718	1228	2066	4478	2245	816	607	458	309	186	272	280	557
1898	13	865	478	396	473	300	71	98	46	180	6	12	36	1022
1898	14	151	111	240	63	190	24	32	14	58	2	4	14	510
1898	15	143	119	130	131	80	70	75	99	91	75	73	93	141
1898	16	220	149	478	252	129	99	71	28	24	16	10	42	194
1898	17	81	27	153	207	72	90	90	81	63	90	99	54	45
1898	18	279	93	527	713	248	310	310	279	217	310	341	186	155
1898	19	540	180	1020	1380	480	600	600	540	420	600	660	360	300
1898	20	992	544	1472	384	288	320	416	416	288	320	192	128	704
1898	21	2108	1156	3128	816	612	680	884	884	612	680	408	272	1496
1898	31	32	-122	-459	-715	-122	49	122	75	94	56	113	132	38
1898	32	102	-10	-168	-617	-292	-25	-32	-7	71	38	47	39	102
1898	33	-117	-80	-18	-75	-35	-14	-7	-1	-2	10	4	-9	-143
1898	34	-27	-22	-39	-13	-15	-5	-6	-3	-3	0	-1	-2	-27
1898	35	-15	-6	-17	-18	-16	-14	-15	-20	-9	3	3	2	-15
1898	36	-16	-21	-18	-24	-26	-20	-4	0	0	0	0	0	-16
1898	37	-8	-3	-15	-21	-7	-9	0	0	0	0	0	0	-5
1898	38	-112	-32	-211	-264	-75	0	0	-21	-32	-32	-32	-74	-62
1898	39	0	-50	-171	-402	[illegible]	[illegible]	0	0	0	0	0	0	0
1898	40	-26	-26	-26	-36	[illegible]	[illegible]	[illegible]	0	0	0	0	-13	-27
1898	41	-111	-123	-125	[illegible]	[illegible]	[illegible]	[illegible]	-116	-63	-28	-27	-46	-111

1898	51	5000	5400	3800	3100	6200	5500	5300	5500	5800				5400
1899	1	2.30		1.79		1.61		1.12		1.51		1.17		EVAP
1899	11	413	746	1411	2563	2181	681	414	413	480	298	309	315	504
1899	12	557	3328	1422	4301	3994	845	451	303	341	268	337	399	1394
1899	13	1022	403	480	614	232	110	239	61	77	-61	-95	28	778
1899	14	510	143	125	179	67	34	75	20	24	-38	-58	12	186
1899	15	141	115	126	127	77	68	73	95	87	73	77	99	149
1899	16	194	157	381	300	91	60	46	107	89	58	30	161	577
1899	17	45	72	333	396	369	234	81	54	54	90	81	36	108
1899	18	155	248	1147	1364	1271	806	279	186	186	310	279	124	372
1899	19	300	480	2220	2640	2460	1560	540	360	360	600	540	240	720
1899	20	704	224	704	352	224	128	288	224	160	160	128	96	832
1899	21	1496	476	1496	748	476	272	612	476	340	340	272	204	1768
1899	31	38	-141	-423	-769	-312	56	141	40	50	30	60	70	20
1899	32	102	-10	-168	-617	-292	-25	-32	-7	71	38	47	39	102
1899	33	-143	-80	-18	-75	-35	-14	-7	-1	-2	77	111	-7	-195
1899	34	-27	-29	-25	-36	-13	-7	-15	-4	-3	0	0	-2	-27
1899	35	-15	-6	-17	-18	-15	-14	-15	-19	-11	3	3	2	-15
1899	36	-16	-21	-18	-24	-18	0	0	-6	-6	0	0	-5	-16
1899	37	-5	-7	-27	-40	-27	-11	0	0	0	0	0	0	-11
1899	38	-62	-99	-229	-246	-75	0	0	-21	-32	-32	-32	-50	-149
1899	39	0	-50	-171	-402	-251	-131	0	0	0	0	0	0	0
1899	40	-27	-22	-30	-35	-4	0	0	0	0	0	0	-10	-30
1899	41	-111	-119	-129	67	104	126	213	-116	-63	-28	-27	-46	-111
1899	51	5400	5100	2600	2100	3200	4000	5500	5900	5800				4600

{additional data deleted}

Executing the program.

To execute the program, you must define the input data file and the output HECDSS data file on the execution line. If you use the "MENUPRM", it will help you in defining the files. From the standpoint of MENUPRM, the file "D0DATA" should be stored on your hard disk with the extension "D0D". To run it from the DOS prompt, you must enter the programs name followed by file assignments. For example, to use the above example file named "RDATA0.IN" and store it an HECDSS data file named "EXAMPLE.DSS", enter the following command at the DOS prompt:

```
RDATA0 D=RDATA0.IN DSS=EXAMPLE
```

Example Output From RDATA0
Period of Record

```
-----DSS---ZOPEN:  Existing File Opened,  File: TSIN.DSS
                   Unit:  71;  DSS Version: 6-GO

Enter year of depletions: 1991
EVAP RECORD:
1898     1       2.30       1.79       1.61       1.12       1.51       1.17       EVAP
51   RECORD:
1898    51       5000 5400 3800 3100 6200 5500 5300 5500 5800                      5400
EVAP RECORD:
1899     1       2.30       1.79       1.61       1.12       1.51       1.17       EVAP
51   RECORD:
1899    51       5400 5100 2600 2100 3200 4000 5500 5900 5800                      4600
EVAP RECORD:
1900     1       2.30       1.79       1.61       1.12       1.51       1.17       EVAP
51   RECORD:
1900    51       4600 4500 3500 3500 5000 4800 4800 5000 6000                      3800
EVAP RECORD:
1901     1       2.30       1.79       1.61       1.12       1.51       1.17       EVAP
51   RECORD:
1901    51       3800 2800 5300 3400 6200 6200 5900 5100 5000                      3800
EVAP RECORD:
1902     1       2.30       1.79       1.61       1.12       1.51       1.17       EVAP
51   RECORD:
1902    51       3800 3600 3700 2300 2700 4400 4100 3800 5200                      3200
EVAP RECORD:
1903     1       2.30       1.79       1.61       1.12       1.51       1.17       EVAP
51   RECORD:
1903    51       3200 4200 2500 2100 4500 3400 3400 4600 5200                      4400
EVAP RECORD:
1904     1       2.30       1.79       1.61       1.12       1.51       1.17       EVAP
51   RECORD:
1904    51       4400 3000 2500 2100 2900 4700 5300 5300 5000                      3200
EVAP RECORD:
1905     1       2.30       1.79       1.61       1.12       1.51       1.17       EVAP
51   RECORD:
1905    51       3200 3600 2800 2500 3400 4200 4000 4700 5400                      4600
EVAP RECORD:
1906     1       2.30       1.79       1.61       1.12       1.51       1.17       EVAP
51   RECORD:
1906    51       4600 3000 3900 4200 5400 5200 5500 5400 4800                      4200
EVAP RECORD:
1907     1       2.30       1.79       1.61       1.12       1.51       1.17       EVAP
51   RECORD:
1907    51       4200 4300 5800 2400 3000 3400 5200 5900 5200                      4000
EVAP RECORD:
1908     1       2.30       1.79       1.61       1.12       1.51       1.17       EVAP
51   RECORD:
1908    51       4000 5400 3900 2100 3300 4400 5200 5200 6000                      3200
```

{additional data deleted}

```
EVAP RECORD:
1984     1       3.62       2.72       2.60       3.06       2.60       3.17       EVAP
51   RECORD:
1984    51       1800 2600 2100 3000 2900 6200 6000 6200 6000                      6200
EVAP RECORD:
1985     1       3.28       2.40       2.65       3.08       2.84       2.49       EVAP
51   RECORD:
1985    51       6200 1900 6200 6000 6000 6200 6000 6200 6000                      6200
EVAP RECORD:
1986     1       2.81       2.20       2.11       2.62       2.52       2.16       EVAP
51   RECORD:
1986    51       6200 2300 2900 2700 2500 2432 2800 2900 2800                      6200
EVAP RECORD:
1987     1       2.77       2.44       2.51          3       2.68       2.70       EVAP
51   RECORD:
1987    51       1839 2272 1982 6000 6200 1939 6200 6200 6200                      6200
EVAP RECORD:
1988     1       3.49       2.63       2.97          4       3.03       3.21       EVAP
51   RECORD:
1988    51       6200 6000 6200 6000 6200 6200 6200 6200 6000                      6200
EVAP RECORD:
1989     1       3.48       2.93       3.29          3       3.19       3.50       EVAP
51   RECORD:
1989    51       6200 6000 6200 6000 6200 6200 6200 6200 6000                      6200

Identification   0 not found!   (real    0)
-----DSS---ZWRITE Unit  71; Vers.    4:  //FTPK/EVAP_RATE/01JAN1890/1MON//
-----DSS---ZWRITE Unit  71; Vers.    4:  //FTPK/EVAP_RATE/01JAN1900/1MON//
-----DSS---ZWRITE Unit  71; Vers.    4:  //FTPK/EVAP_RATE/01JAN1910/1MON//
```

```
-----DSS---ZWRITE Unit  71; Vers.    4:  //FTPK/EVAP_RATE/01JAN1920/1MON//
-----DSS---ZWRITE Unit  71; Vers.    4:  //FTPK/EVAP_RATE/01JAN1930/1MON//
-----DSS---ZWRITE Unit  71; Vers.    4:  //FTPK/EVAP_RATE/01JAN1940/1MON//
-----DSS---ZWRITE Unit  71; Vers.    4:  //FTPK/EVAP_RATE/01JAN1950/1MON//
-----DSS---ZWRITE Unit  71; Vers.    4:  //FTPK/EVAP_RATE/01JAN1960/1MON//
-----DSS---ZWRITE Unit  71; Vers.    4:  //FTPK/EVAP_RATE/01JAN1970/1MON//
-----DSS---ZWRITE Unit  71; Vers.    4:  //FTPK/EVAP_RATE/01JAN1980/1MON//
-----DSS---ZWRITE Unit  71; Vers.    4:  //FTPK/EVAP_RATE/01JAN1990/1MON//
-----DSS---ZWRITE Unit  71; Vers.    4:  //FTPK/EVAP_RATE/01JAN1800/1YEAR//
-----DSS---ZWRITE Unit  71; Vers.    4:  //FTPK/EVAP_RATE/01JAN1900/1YEAR//
-----DSS---ZWRITE Unit  71; Vers.    4:  //GARR/EVAP_RATE/01JAN1890/1MON//
-----DSS---ZWRITE Unit  71; Vers.    4:  //GARR/EVAP_RATE/01JAN1900/1MON//
-----DSS---ZWRITE Unit  71; Vers.    4:  //GARR/EVAP_RATE/01JAN1910/1MON//
```

{additional data deleted}

```
-----DSS---ZWRITE Unit  71; Vers.    4:  //GARR/FLOW_LOC_INC/01JAN1980/1MON//
-----DSS---ZWRITE Unit  71; Vers.    4:  //GARR/FLOW_LOC_INC/01JAN1990/1MON//
-----DSS---ZWRITE Unit  71; Vers.    4:  //OAHE/FLOW_LOC_INC/01JAN1890/1MON//
-----DSS---ZWRITE Unit  71; Vers.    4:  //OAHE/FLOW_LOC_INC/01JAN1900/1MON//
-----DSS---ZWRITE Unit  71; Vers.    4:  //OAHE/FLOW_LOC_INC/01JAN1910/1MON//
-----DSS---ZWRITE Unit  71; Vers.    4:  //OAHE/FLOW_LOC_INC/01JAN1920/1MON//
-----DSS---ZWRITE Unit  71; Vers.    4:  //OAHE/FLOW_LOC_INC/01JAN1930/1MON//
-----DSS---ZWRITE Unit  71; Vers.    4:  //OAHE/FLOW_LOC_INC/01JAN1940/1MON//
-----DSS---ZWRITE Unit  71; Vers.    4:  //OAHE/FLOW_LOC_INC/01JAN1950/1MON//
-----DSS---ZWRITE Unit  71; Vers.    4:  //OAHE/FLOW_LOC_INC/01JAN1960/1MON//
-----DSS---ZWRITE Unit  71; Vers.    4:  //OAHE/FLOW_LOC_INC/01JAN1970/1MON//
-----DSS---ZWRITE Unit  71; Vers.    4:  //OAHE/FLOW_LOC_INC/01JAN1980/1MON//
-----DSS---ZWRITE Unit  71; Vers.    4:  //OAHE/FLOW_LOC_INC/01JAN1990/1MON//
-----DSS---ZWRITE Unit  71; Vers.    4:  //FTRA/FLOW_LOC_INC/01JAN1890/1MON//
-----DSS---ZWRITE Unit  71; Vers.    4:  //FTRA/FLOW_LOC_INC/01JAN1900/1MON//
-----DSS---ZWRITE Unit  71; Vers.    4:  //FTRA/FLOW_LOC_INC/01JAN1910/1MON//
-----DSS---ZWRITE Unit  71; Vers.    4:  //FTRA/FLOW_LOC_INC/01JAN1920/1MON//
-----DSS---ZWRITE Unit  71; Vers.    4:  //FTRA/FLOW_LOC_INC/01JAN1930/1MON//
-----DSS---ZWRITE Unit  71; Vers.    4:  //FTRA/FLOW_LOC_INC/01JAN1940/1MON//
-----DSS---ZWRITE Unit  71; Vers.    4:  //FTRA/FLOW_LOC_INC/01JAN1950/1MON//
-----DSS---ZWRITE Unit  71; Vers.    4:  //FTRA/FLOW_LOC_INC/01JAN1960/1MON//
-----DSS---ZWRITE Unit  71; Vers.    4:  //FTRA/FLOW_LOC_INC/01JAN1970/1MON//
-----DSS---ZWRITE Unit  71; Vers.    4:  //FTRA/FLOW_LOC_INC/01JAN1980/1MON//
-----DSS---ZWRITE Unit  71; Vers.    4:  //FTRA/FLOW_LOC_INC/01JAN1990/1MON//
-----DSS---ZWRITE Unit  71; Vers.    4:  //GAPT/FLOW_LOC_INC/01JAN1890/1MON//
-----DSS---ZWRITE Unit  71; Vers.    4:  //GAPT/FLOW_LOC_INC/01JAN1900/1MON//
-----DSS---ZWRITE Unit  71; Vers.    4:  //GAPT/FLOW_LOC_INC/01JAN1910/1MON//
-----DSS---ZWRITE Unit  71; Vers.    4:  //GAPT/FLOW_LOC_INC/01JAN1920/1MON//
-----DSS---ZWRITE Unit  71; Vers.    4:  //GAPT/FLOW_LOC_INC/01JAN1930/1MON//
-----DSS---ZWRITE Unit  71; Vers.    4:  //GAPT/FLOW_LOC_INC/01JAN1940/1MON//
-----DSS---ZWRITE Unit  71; Vers.    4:  //GAPT/FLOW_LOC_INC/01JAN1950/1MON//
-----DSS---ZWRITE Unit  71; Vers.    4:  //GAPT/FLOW_LOC_INC/01JAN1960/1MON//
-----DSS---ZWRITE Unit  71; Vers.    4:  //GAPT/FLOW_LOC_INC/01JAN1970/1MON//
-----DSS---ZWRITE Unit  71; Vers.    4:  //GAPT/FLOW_LOC_INC/01JAN1980/1MON//
-----DSS---ZWRITE Unit  71; Vers.    4:  //GAPT/FLOW_LOC_INC/01JAN1990/1MON//
-----DSS---ZWRITE Unit  71; Vers.    4:  //SUX/FLOW_LOC_INC/01JAN1890/1MON//
-----DSS---ZWRITE Unit  71; Vers.    4:  //SUX/FLOW_LOC_INC/01JAN1900/1MON//
-----DSS---ZWRITE Unit  71; Vers.    4:  //SUX/FLOW_LOC_INC/01JAN1910/1MON//
-----DSS---ZWRITE Unit  71; Vers.    4:  //SUX/FLOW_LOC_INC/01JAN1920/1MON//
-----DSS---ZWRITE Unit  71; Vers.    4:  //SUX/FLOW_LOC_INC/01JAN1930/1MON//
-----DSS---ZWRITE Unit  71; Vers.    4:  //SUX/FLOW_LOC_INC/01JAN1940/1MON//
-----DSS---ZWRITE Unit  71; Vers.    4:  //SUX/FLOW_LOC_INC/01JAN1950/1MON//
-----DSS---ZWRITE Unit  71; Vers.    4:  //SUX/FLOW_LOC_INC/01JAN1960/1MON//
```

{additional data deleted}

```
-----DSS---ZWRITE Unit  71; Vers.    4:  //HEMO/FLOW_LOC_INC/01JAN1970/1MON//
-----DSS---ZWRITE Unit  71; Vers.    4:  //HEMO/FLOW_LOC_INC/01JAN1980/1MON//
-----DSS---ZWRITE Unit  71; Vers.    4:  //HEMO/FLOW_LOC_INC/01JAN1990/1MON//
-----DSS---ZWRITE Unit  71; Vers.    1:  //FTPK/FLOW_LOC_DPL/01JAN1890/1MON/1991/
-----DSS---ZWRITE Unit  71; Vers.    1:  //FTPK/FLOW_LOC_DPL/01JAN1900/1MON/1991/
-----DSS---ZWRITE Unit  71; Vers.    1:  //FTPK/FLOW_LOC_DPL/01JAN1910/1MON/1991/
-----DSS---ZWRITE Unit  71; Vers.    1:  //FTPK/FLOW_LOC_DPL/01JAN1920/1MON/1991/
-----DSS---ZWRITE Unit  71; Vers.    1:  //FTPK/FLOW_LOC_DPL/01JAN1930/1MON/1991/
-----DSS---ZWRITE Unit  71; Vers.    1:  //FTPK/FLOW_LOC_DPL/01JAN1940/1MON/1991/
-----DSS---ZWRITE Unit  71; Vers.    1:  //FTPK/FLOW_LOC_DPL/01JAN1950/1MON/1991/
-----DSS---ZWRITE Unit  71; Vers.    1:  //FTPK/FLOW_LOC_DPL/01JAN1960/1MON/1991/
-----DSS---ZWRITE Unit  71; Vers.    1:  //FTPK/FLOW_LOC_DPL/01JAN1970/1MON/1991/
-----DSS---ZWRITE Unit  71; Vers.    1:  //FTPK/FLOW_LOC_DPL/01JAN1980/1MON/1991/
-----DSS---ZWRITE Unit  71; Vers.    1:  //FTPK/FLOW_LOC_DPL/01JAN1990/1MON/1991/
-----DSS---ZWRITE Unit  71; Vers.    1:  //GARR/FLOW_LOC_DPL/01JAN1890/1MON/1991/
-----DSS---ZWRITE Unit  71; Vers.    1:  //GARR/FLOW_LOC_DPL/01JAN1900/1MON/1991/
-----DSS---ZWRITE Unit  71; Vers.    1:  //GARR/FLOW_LOC_DPL/01JAN1910/1MON/1991/
-----DSS---ZWRITE Unit  71; Vers.    1:  //GARR/FLOW_LOC_DPL/01JAN1920/1MON/1991/
-----DSS---ZWRITE Unit  71; Vers.    1:  //GARR/FLOW_LOC_DPL/01JAN1930/1MON/1991/
-----DSS---ZWRITE Unit  71; Vers.    1:  //GARR/FLOW_LOC_DPL/01JAN1940/1MON/1991/
-----DSS---ZWRITE Unit  71; Vers.    1:  //GARR/FLOW_LOC_DPL/01JAN1950/1MON/1991/
-----DSS---ZWRITE Unit  71; Vers.    1:  //GARR/FLOW_LOC_DPL/01JAN1960/1MON/1991/
```

```
-----DSS---ZWRITE Unit  71; Vers.    1:  //GARR/FLOW_LOC_DPL/01JAN1970/1MON/1991/
-----DSS---ZWRITE Unit  71; Vers.    1:  //GARR/FLOW_LOC_DPL/01JAN1980/1MON/1991/
-----DSS---ZWRITE Unit  71; Vers.    1:  //GARR/FLOW_LOC_DPL/01JAN1990/1MON/1991/
-----DSS---ZWRITE Unit  71; Vers.    1:  //OAHE/FLOW_LOC_DPL/01JAN1890/1MON/1991/
-----DSS---ZWRITE Unit  71; Vers.    1:  //OAHE/FLOW_LOC_DPL/01JAN1900/1MON/1991/
-----DSS---ZWRITE Unit  71; Vers.    1:  //OAHE/FLOW_LOC_DPL/01JAN1910/1MON/1991/
-----DSS---ZWRITE Unit  71; Vers.    1:  //OAHE/FLOW_LOC_DPL/01JAN1920/1MON/1991/
-----DSS---ZWRITE Unit  71; Vers.    1:  //OAHE/FLOW_LOC_DPL/01JAN1930/1MON/1991/
-----DSS---ZWRITE Unit  71; Vers.    1:  //OAHE/FLOW_LOC_DPL/01JAN1940/1MON/1991/
-----DSS---ZWRITE Unit  71; Vers.    1:  //OAHE/FLOW_LOC_DPL/01JAN1950/1MON/1991/
-----DSS---ZWRITE Unit  71; Vers.    1:  //OAHE/FLOW_LOC_DPL/01JAN1960/1MON/1991/
-----DSS---ZWRITE Unit  71; Vers.    1:  //OAHE/FLOW_LOC_DPL/01JAN1970/1MON/1991/
-----DSS---ZWRITE Unit  71; Vers.    1:  //OAHE/FLOW_LOC_DPL/01JAN1980/1MON/1991/
-----DSS---ZWRITE Unit  71; Vers.    1:  //OAHE/FLOW_LOC_DPL/01JAN1990/1MON/1991/
-----DSS---ZWRITE Unit  71; Vers.    1:  //FTRA/FLOW_LOC_DPL/01JAN1890/1MON/1991/
-----DSS---ZWRITE Unit  71; Vers.    1:  //FTRA/FLOW_LOC_DPL/01JAN1900/1MON/1991/
-----DSS---ZWRITE Unit  71; Vers.    1:  //FTRA/FLOW_LOC_DPL/01JAN1910/1MON/1991/
-----DSS---ZWRITE Unit  71; Vers.    1:  //FTRA/FLOW_LOC_DPL/01JAN1920/1MON/1991/
-----DSS---ZWRITE Unit  71; Vers.    1:  //FTRA/FLOW_LOC_DPL/01JAN1930/1MON/1991/
-----DSS---ZWRITE Unit  71; Vers.    1:  //FTRA/FLOW_LOC_DPL/01JAN1940/1MON/1991/
-----DSS---ZWRITE Unit  71; Vers.    1:  //FTRA/FLOW_LOC_DPL/01JAN1950/1MON/1991/
-----DSS---ZWRITE Unit  71; Vers.    1:  //FTRA/FLOW_LOC_DPL/01JAN1960/1MON/1991/
-----DSS---ZWRITE Unit  71; Vers.    1:  //FTRA/FLOW_LOC_DPL/01JAN1970/1MON/1991/
-----DSS---ZWRITE Unit  71; Vers.    1:  //FTRA/FLOW_LOC_DPL/01JAN1980/1MON/1991/
-----DSS---ZWRITE Unit  71; Vers.    1:  //FTRA/FLOW_LOC_DPL/01JAN1990/1MON/1991/
-----DSS---ZWRITE Unit  71; Vers.    1:  //GAPT/FLOW_LOC_DPL/01JAN1890/1MON/1991/
```

{additional data deleted}

```
-----DSS---ZWRITE Unit  71; Vers.    1:  //BNMO/FLOW_LOC_DPL/01JAN1970/1MON/1991/
-----DSS---ZWRITE Unit  71; Vers.    1:  //BNMO/FLOW_LOC_DPL/01JAN1980/1MON/1991/
-----DSS---ZWRITE Unit  71; Vers.    1:  //BNMO/FLOW_LOC_DPL/01JAN1990/1MON/1991/
-----DSS---ZWRITE Unit  71; Vers.    1:  //HEMO/FLOW_LOC_DPL/01JAN1890/1MON/1991/
-----DSS---ZWRITE Unit  71; Vers.    1:  //HEMO/FLOW_LOC_DPL/01JAN1900/1MON/1991/
-----DSS---ZWRITE Unit  71; Vers.    1:  //HEMO/FLOW_LOC_DPL/01JAN1910/1MON/1991/
-----DSS---ZWRITE Unit  71; Vers.    1:  //HEMO/FLOW_LOC_DPL/01JAN1920/1MON/1991/
-----DSS---ZWRITE Unit  71; Vers.    1:  //HEMO/FLOW_LOC_DPL/01JAN1930/1MON/1991/
-----DSS---ZWRITE Unit  71; Vers.    1:  //HEMO/FLOW_LOC_DPL/01JAN1940/1MON/1991/
-----DSS---ZWRITE Unit  71; Vers.    1:  //HEMO/FLOW_LOC_DPL/01JAN1950/1MON/1991/
-----DSS---ZWRITE Unit  71; Vers.    1:  //HEMO/FLOW_LOC_DPL/01JAN1960/1MON/1991/
-----DSS---ZWRITE Unit  71; Vers.    1:  //HEMO/FLOW_LOC_DPL/01JAN1970/1MON/1991/
-----DSS---ZWRITE Unit  71; Vers.    1:  //HEMO/FLOW_LOC_DPL/01JAN1980/1MON/1991/
-----DSS---ZWRITE Unit  71; Vers.    1:  //HEMO/FLOW_LOC_DPL/01JAN1990/1MON/1991/
   -----DSS---ZCLOSE Unit:  71,   File: TSIN.DSS
              Pointer Utilization:  0.44
              Number of Records:    695
              File Size:    668.9  Kbytes
              Percent Inactive:   0.0
```

Description of Location Codes in the Data File DODATA

Line ID	Item	Description
1	Evaporation	Annual evaporation in feet beginning in March. (e.g. 2.30 feet for Ft. Peck in Mar 1898)
11-21	Inflow	Inflow to projects in 1,000 Acre-Feet beginning in March. (e.g. 419 KAcre-Feet inflow to Ft. Peck in March 1898).
11		Ft. Peck Inflow (FTPK)
12		Garrison Inflow (GARR)
13		Oahe Inflow (OAHE)
14		Ft. Randall & Big Bend Inflow (FTRA)
15		Gavins Point Inflow (GAPT)
16		Sioux City Inflow (SUX)
17		Omaha Inflow (OMA)
18		Nebraska City Inflow (NCNE)
19		Kansas City Inflow (MKC)
20		Boonville Inflow (BNMO)
21		Hermann Inflow (HEMO)
21-41	Depletions	Depletions from projects in 1,000 Acre-Feet beginning in March. (e.g. 32 KAcre-Feet depletion from Ft. Peck in March 1898). The include diversions.
31		Ft. Peck Depletion (FTPK)
32		Garrison Depletion (GARR)
33		Oahe Depletion (OAHE)
34		Ft. Randall & Big Bend Depletion (FTRA)
35		Gavins Point Depletion (GAPT)
36		Sioux City Depletion (SUX)
37		Omaha Depletion (OMA)
38		Nebraska City Depletion (NCNE)
39		Kansas City Depletion (MKC)
40		Boonville Depletion (BNMO)
41		Hermann Depletion (HEMO)
51	Limit	Upper limit on releases from Gavin's Point.

Evaporation Rates

Distribution of evaporation

Standard Time Period	Beginning Period Time	End-of-Period Time	Array T15 Row #	Evaporation Distribution (in percent)
1	16MAR	22MAR	1	0
2	23MAR	31MAR	2	0
3	01APR	30APR	3	0
4	01MAY	31MAY	4	7
5	01JUN	30JUN	5	5
6	01JUL	31JUL	6	19
7	01AUG	31AUG	7	20
8	01SEP	30SEP	8	19
9	01OCT	31OCT	9	13
10	01NOV	15NOV	10	6
11	16NOV	30NOV	11	6
12	01DEC	31DEC	12	5
13	01JAN	31JAN	13	0
14	01FEB	28FEB	14	0
15	01MAR	15MAR	15	0

Listing of Computer Source Code for RDATA0

```
      PROGRAM RDDATA0
C
C     Read data from MRD, d0data file
C     -----------------------------------------------------------------
      include 'rdata0.cbl'
C     -----------------------------------------------------------------
C
      CALL ATTACH(5,'INPUT','STDIN',' ',CFNAME,IOER01)
      CALL ATTACH(6,'OUTPUT','STDOUT',' ',CFNAME,IOER03)
C
      CINREC='D0DATA'
      CALL CHRLNB(CINREC,NINREC)
      CALL ATTACH(1,'D0DATA',CINREC(1:NINREC),' ',CFNAME,IOER02)
C
      CINREC='MRD.DSS'
      CALL CHRLNB(CINREC,NINREC)
      CALL ATTACH(71,'DSSFILE',CINREC(1:NINREC),'NOP',CFNDSS,IOER04)
      CALL ATTEND
C
      CALL ZSET('PROG','RDATA0',0)
      CALL ZOPEN(IFLTAB,CFNDSS,IERR01)
C
      CALL RD_USRI ()
C
      DO 1400 IULOC=1,NULOC
      DO 1400 JP=1,6
      CALL CHRLNB(CP(JP,IULOC),NP(JP,IULOC))
 1400 CONTINUE
C
C     Read Records of time series data and store data in DSS data file
C     _________________________________________________________________
      CALL RD_D0DATA ()
C
 9000 CALL ZCLOSE(IFLTAB)
      STOP
      END
```

```
      BLOCK DATA
C
      include 'rdata0.cbl'
C
      DATA NULOC/22/
C
      DATA IDENT(1)/11/, CP(1,1)/'                  '/, CP(2,1)/'FTPK'/,
     .CP(3,1)/'FLOW_LOC_INC'/, CP(5,1)/'1MON'/, CP(6,1)/'        '/
      DATA IDENT(2)/12/, CP(1,2)/'                  '/, CP(2,2)/'GARR'/,
     .CP(3,2)/'FLOW_LOC_INC'/, CP(5,2)/'1MON'/, CP(6,2)/'        '/
      DATA IDENT(3)/13/, CP(1,3)/'                  '/, CP(2,3)/'OAHE'/,
     .CP(3,3)/'FLOW_LOC_INC'/, CP(5,3)/'1MON'/, CP(6,3)/'        '/
      DATA IDENT(4)/14/, CP(1,4)/'                  '/, CP(2,4)/'FTRA'/,
     .CP(3,4)/'FLOW_LOC_INC'/, CP(5,4)/'1MON'/, CP(6,4)/'        '/
      DATA IDENT(5)/15/, CP(1,5)/'                  '/, CP(2,5)/'GAPT'/,
     .CP(3,5)/'FLOW_LOC_INC'/, CP(5,5)/'1MON'/, CP(6,5)/'        '/
      DATA IDENT(6)/16/, CP(1,6)/'                  '/, CP(2,6)/'SUX'/,
     .CP(3,6)/'FLOW_LOC_INC'/, CP(5,6)/'1MON'/, CP(6,6)/'        '/
      DATA IDENT(7)/17/, CP(1,7)/'                  '/, CP(2,7)/'OMA '/,
     .CP(3,7)/'FLOW_LOC_INC'/, CP(5,7)/'1MON'/, CP(6,7)/'        '/
      DATA IDENT(8)/18/, CP(1,8)/'                  '/, CP(2,8)/'NCNE'/,
     .CP(3,8)/'FLOW_LOC_INC'/, CP(5,8)/'1MON'/, CP(6,8)/'        '/
      DATA IDENT(9)/19/, CP(1,9)/'                  '/, CP(2,9)/'MKC '/,
     .CP(3,9)/'FLOW_LOC_INC'/, CP(5,9)/'1MON'/, CP(6,9)/'        '/
      DATA IDENT(10)/20/, CP(1,10)/'                  '/, CP(2,10)/'BNMO'/,
     .CP(3,10)/'FLOW_LOC_INC'/, CP(5,10)/'1MON'/, CP(6,10)/'        '/
      DATA IDENT(11)/21/, CP(1,11)/'                  '/, CP(2,11)/'HEMO'/,
     .CP(3,11)/'FLOW_LOC_INC'/, CP(5,11)/'1MON'/, CP(6,11)/'        '/
C
      DATA MX_LOC_DPL/12,22/
      DATA IDENT(12)/31/, CP(1,12)/'                  '/, CP(2,12)/'FTPK'/,
     .CP(3,12)/'FLOW_LOC_DPL'/, CP(5,12)/'1MON'/, CP(6,12)/'        '/
      DATA IDENT(13)/32/, CP(1,13)/'                  '/, CP(2,13)/'GARR'/,
     .CP(3,13)/'FLOW_LOC_DPL'/, CP(5,13)/'1MON'/, CP(6,13)/'        '/
      DATA IDENT(14)/33/, CP(1,14)/'                  '/, CP(2,14)/'OAHE'/,
     .CP(3,14)/'FLOW_LOC_DPL'/, CP(5,14)/'1MON'/, CP(6,14)/'        '/
      DATA IDENT(15)/34/, CP(1,15)/'                  '/, CP(2,15)/'FTRA'/,
     .CP(3,15)/'FLOW_LOC_DPL'/, CP(5,15)/'1MON'/, CP(6,15)/'        '/
      DATA IDENT(16)/35/, CP(1,16)/'                  '/, CP(2,16)/'GAPT'/,
     .CP(3,16)/'FLOW_LOC_DPL'/, CP(5,16)/'1MON'/, CP(6,16)/'        '/
      DATA IDENT(17)/36/, CP(1,17)/'                  '/, CP(2,17)/'SUX '/,
     .CP(3,17)/'FLOW_LOC_DPL'/, CP(5,17)/'1MON'/, CP(6,17)/'        '/
      DATA IDENT(18)/37/, CP(1,18)/'                  '/, CP(2,18)/'OMA '/,
     .CP(3,18)/'FLOW_LOC_DPL'/, CP(5,18)/'1MON'/, CP(6,18)/'        '/
      DATA IDENT(19)/38/, CP(1,19)/'                  '/, CP(2,19)/'NCNE'/,
     .CP(3,19)/'FLOW_LOC_DPL'/, CP(5,19)/'1MON'/, CP(6,19)/'        '/
      DATA IDENT(20)/39/, CP(1,20)/'                  '/, CP(2,20)/'MKC '/,
     .CP(3,20)/'FLOW_LOC_DPL'/, CP(5,20)/'1MON'/, CP(6,20)/'        '/
      DATA IDENT(21)/40/, CP(1,21)/'                  '/, CP(2,21)/'BNMO'/,
     .CP(3,21)/'FLOW_LOC_DPL'/, CP(5,21)/'1MON'/, CP(6,21)/'        '/
      DATA IDENT(22)/41/, CP(1,22)/'                  '/, CP(2,22)/'HEMO'/,
     .CP(3,22)/'FLOW_LOC_DPL'/, CP(5,22)/'1MON'/, CP(6,22)/'        '/
C
      DATA IDENT(23)/ 1/, CP(1,23)/'                  '/, CP(2,23)/'    '/,
     .CP(3,23)/'EVAP_RATE'/, CP(5,23)/'1MON'/, CP(6,23)/'    '/
C
CCCCC DATA IDENT(@)/11/, CP(1,@)/'                  '/, CP(2,@)/'FTPK'/,
CCCCC.CP(3,@)/'FLOW        '/, CP(5,@)/'1MON'/, CP(6,@)/'XXXX'/
C
      DATA EVDIST/0.,0.,7.,5.,19.,20.,19.,13.,12.,5.,0.,0./
C
      DATA EVRESP(1,1)/'                  '/, EVRESP(2,1)/'FTPK'/,
     .EVRESP(3,1)/'EVAP_RATE'/, EVRESP(5,1)/'1MON'/, EVRESP(6,1)/' '/
      DATA EVRESP(1,2)/'                  '/, EVRESP(2,2)/'GARR'/,
     .EVRESP(3,2)/'EVAP_RATE'/, EVRESP(5,2)/'1MON'/, EVRESP(6,2)/' '/
      DATA EVRESP(1,3)/'                  '/, EVRESP(2,3)/'OAHE'/,
     .EVRESP(3,3)/'EVAP_RATE'/, EVRESP(5,3)/'1MON'/, EVRESP(6,3)/' '/
      DATA EVRESP(1,4)/'                  '/, EVRESP(2,4)/'BEND'/,
     .EVRESP(3,4)/'EVAP_RATE'/, EVRESP(5,4)/'1MON'/, EVRESP(6,4)/' '/
      DATA EVRESP(1,5)/'                  '/, EVRESP(2,5)/'FTRA'/,
     .EVRESP(3,5)/'EVAP_RATE'/, EVRESP(5,5)/'1MON'/, EVRESP(6,5)/' '/
      DATA EVRESP(1,6)/'                  '/, EVRESP(2,6)/'GAPT'/,
     .EVRESP(3,6)/'EVAP_RATE'/, EVRESP(5,6)/'1MON'/, EVRESP(6,6)/' '/
C
      END
```

```
      SUBROUTINE CHRLJS (CIN,COUT)
C
C     Left justify a string
C     -------------------------------------------------------------
      PARAMETER (KTEMP=132)
      CHARACTER*(*) CIN,COUT,CTEMP*(KTEMP)
C     -------------------------------------------------------------
      KIN=LEN(CIN)
      KOUT=LEN(COUT)
      CALL LFLNB(CIN,1,KIN,IFR,ILEN)
      ILEN=MIN(ILEN,KOUT,KTEMP)
C
      IF(ILEN.LT.1) THEN
        CALL CHRBLK(COUT)
C
      ELSE
        CTEMP=CIN
        ILAST=IFR+ILEN-1
        DO 10 I=1,KOUT
        ICOL=IFR+I-1
        IF(ICOL.LE.KIN) THEN
          COUT(I:I)=CTEMP(ICOL:ICOL)
        ELSE
          COUT(I:I)=' '
        ENDIF
   10   CONTINUE
      ENDIF
C
      RETURN
      END
```

```
      SUBROUTINE RD_D0DATA ()
C
C     ----------------------------------------------------------------
      INTEGER PREV_YEAR, IYEAR, JIDENT
C
      include 'rdata0.cbl'
C     ----------------------------------------------------------------
      IX_YEAR=0
      PREV_YEAR=-9999
      BASE_YR=' '
C
C     Read a Record of time series data
C     ----------------------------------------------------------------
 1500 READ(1,1505,END=5000) CINREC
 1505 FORMAT(A)
C
      CALL CHRLNB(CINREC,NPRCOL)
      NPRCOL=MAX(1,NPRCOL)
C
      IF(CINREC(9:10).EQ.'51') THEN
        WRITE(6,'('' 51   RECORD:'',/1X,A)') CINREC(1:NPRCOL)
      ELSE
        IF(CINREC(77:80).EQ.'EVAP') THEN
          WRITE(6,'('' EVAP RECORD:'',/1X,A)') CINREC(1:NPRCOL)
        ENDIF
C
        READ(CINREC(1:10),'(I4,4X,I2)') IYEAR,JIDENT
C
        IXLOC=0
        IULOC=1
        DO WHILE(IULOC.LE.NULOC)
          IF(JIDENT.EQ.IDENT(IULOC)) THEN
            IXLOC=IULOC
            GO TO 2000
          ENDIF
          IULOC=IULOC+1
        END DO
C
        IF(JIDENT.EQ.1) THEN
          IXLOC=NULOC+1
        ELSE
          WRITE(6,1520) JIDENT,IYEAR
 1520     FORMAT(/' Identification ',I3,' not found!   (Year',I5,')')
          GO TO 1500
        ENDIF
C
 2000   IF(IYEAR.NE.PREV_YEAR) THEN
          IF(IX_YEAR+1.GT.KYEAR) THEN
            CALL WR_EVAP ()
            CALL WR_FLOW ()
            IX_YEAR=1
            BASE_YR=' '
          ELSE
            IX_YEAR=IX_YEAR+1
          ENDIF
        ENDIF
        PREV_YEAR=IYEAR
        IF(BASE_YR.EQ.' ') BASE_YR=CINREC(1:4)
C
C       evaporation (yearly)
C     ----------------------------------------------------------------
        IF(JIDENT.EQ.1) THEN
          READ(CINREC,1600) (EVRATE(J),J=1,6)
 1600     FORMAT(10X,6F10.0)
          JLOC=1
          DO WHILE(JLOC.LE.6)
            VAL_YR(IX_YEAR,JLOC)=EVRATE(JLOC)
            JLOC=JLOC+1
          END DO
C
C         Distribute yearly over the month
C
          JXLOC=1
          DO WHILE (JXLOC.LE.6)
            JMO=1
            DO WHILE (JMO.LE.12)
              VALUES(JMO)=EVRATE(JXLOC)*EVDIST(JMO)*0.01
              VAL_MON((IX_YEAR-1)*12+JMO,NULOC+JXLOC)=EVRATE(JXLOC)*
     *        EVDIST(JMO)*0.01
              JMO=JMO+1
            END DO
            JXLOC=JXLOC+1
          END DO
C
C
C       Inflow or Depletion
C     ----------------------------------------------------------------
        ELSE
          JXYEAR=(IX_YEAR-1)*12
          READ(CINREC,1700) (VALUES(J),J=1,12)
 1700     FORMAT(15X,12F5.0)
```

```
          JMO=1
          DO WHILE(JMO.LE.12)
            VAL_MON(JXYEAR+JMO,IXLOC)=VALUES(JMO)
            JMO=JMO+1
          END DO
        ENDIF
C
      ENDIF
      GO TO 1500
C
 5000 IF(IX_YEAR.GT.0) THEN
        CALL WR_EVAP ()
        CALL WR_FLOW ()
C
        IX_YEAR=0
      ENDIF
C
C     ------------------------------------------------------------------
      RETURN
      END
```

```
      SUBROUTINE RD_USRI ()
C
      include 'rdata0.cbl'
C
      ID_DPL_ALT=' '
      NC_ID_DPL_ALT=0
C
 1500 WRITE(6,1505)
 1505 FORMAT(/' Enter year of depletions: ')
      READ(5,1510,ERR=9000,END=9000) CINREC
 1510 FORMAT(A)
      CALL LFLNB(CINREC,1,80,IFR,ILEN)
      IF(ILEN.GT.0) THEN
        ID_DPL_ALT=CINREC(IFR:IFR+ILEN-1)
        CALL CHRLNB(ID_DPL_ALT,NC_ID_DPL_ALT)
      ELSE
        ID_DPL_ALT=' '
        NC_ID_DPL_ALT=0
      ENDIF
C
C     Store year of depletions in part F of depletion pathname
C
      IXLOC=1
      DO WHILE(IXLOC.LE.KULOC)
        IF(IXLOC.GE.MX_LOC_DPL(1) .AND. IXLOC.LE.MX_LOC_DPL(2)) THEN
          CALL CHRLJS(ID_DPL_ALT,CP(6,IXLOC))
        ENDIF
        IXLOC=IXLOC+1
      END DO
C
 9000 RETURN
      END
```

```
      SUBROUTINE WR_EVAP ()
C
C     Store evaporation data in DSS file
C
      LOGICAL LFIRST
      include 'rdata0.cbl'
C
      DATA LFIRST/.TRUE./
C
C
C
C     Setup Pathname Parts
C
      IF(LFIRST) THEN
        LFIRST=.FALSE.
        JULOC=1
        DO WHILE (JULOC.LE.6)
          JP=1
          DO WHILE (JP.LE.6)
            CALL CHRLNB(EVRESP(JP,JULOC),NVRESP(JP,JULOC))
            JP=JP+1
          END DO
          JULOC=JULOC+1
        END DO
      ENDIF
C
C     Store KYEAR's data
C
      JXLOC=1
      DO WHILE (JXLOC.LE.6)
C
        CALL ZFPN(EVRESP(1,JXLOC),NVRESP(1,JXLOC),EVRESP(2,JXLOC),
     .  NVRESP(2,JXLOC),EVRESP(3,JXLOC),NVRESP(3,JXLOC),EVRESP(4,JXLOC),
     .  NVRESP(4,JXLOC),EVRESP(5,JXLOC),NVRESP(5,JXLOC),EVRESP(6,JXLOC),
     .  NVRESP(6,JXLOC),CPATH,NPATH)
C
        CDATE='31MAR'//BASE_YR
        CTIME='2400'
        NVALS=IX_YEAR*12
        CUNITS='FT       '
        CTYPE='PER-CUM '
        IPLAN=0
C
        CALL ZSRTS(IFLTAB,CPATH,CDATE,CTIME,NVALS,
     .  VAL_MON(1,NULOC+JXLOC),CUNITS,CTYPE,IPLAN,ISTAT)
C
C
C       Yearly interval evaporation
C
        NVALS=IX_YEAR
        CDATE='31DEC'//BASE_YR
        CUNITS='FT/YR    '
        IPLAN=0
C
        CALL ZFPN(EVRESP(1,JXLOC),NVRESP(1,JXLOC),EVRESP(2,JXLOC),
     .  NVRESP(2,JXLOC),EVRESP(3,JXLOC),NVRESP(3,JXLOC),EVRESP(4,JXLOC),
     .  NVRESP(4,JXLOC),'1YEAR',5,EVRESP(6,JXLOC),
     .  NVRESP(6,JXLOC),CPATH,NPATH)
C
        CALL ZSRTS(IFLTAB,CPATH,CDATE,CTIME,NVALS,VAL_YR(1,JXLOC),
     .  CUNITS,CTYPE,IPLAN,ISTAT)
C
        JXLOC=JXLOC+1
      END DO
C
      RETURN
      END
```

```
      SUBROUTINE WR_FLOW ()
C
C     Write Inflows and Depletions to DSS file
C     ----------------------------------------------------------
      include 'rdata0.cbl'
C     ----------------------------------------------------------
C
      IXLOC=1
      DO WHILE(IXLOC.LE.NULOC)
        CALL ZFPN(CP(1,IXLOC),NP(1,IXLOC),CP(2,IXLOC),NP(2,IXLOC),
     .  CP(3,IXLOC),NP(3,IXLOC),' ',0,CP(5,IXLOC),NP(5,IXLOC),
     .  CP(6,IXLOC),NP(6,IXLOC),CPATH,NPATH)
C
        CDATE='31MAR'//BASE_YR
        CTIME='2400'
        NVALS=12*IX_YEAR
        CUNITS='KAF     '
        CTYPE='PER-AVER'
        IPLAN=0
C
        CALL ZSRTS(IFLTAB,CPATH,CDATE,CTIME,NVALS,VAL_MON(1,IXLOC),
     .  CUNITS,CTYPE,IPLAN,ISTAT)
C
        IXLOC=IXLOC+1
      END DO
C
      RETURN
      END
```

```
      PARAMETER(KULOC=30, KYEAR=125)
C
      CHARACTER CP(6,KULOC)*32,CFNAME*64,CFNDSS*64,CINREC*132,
     . CPATH*80,CDATE*9,CTIME*4,CUNITS*8,CTYPE*8,EVRESP(6,6)*32,
     . ID_DPL_ALT*10, BASE_YR*4
C
      COMMON /CAA/ CP,CFNAME,CFNDSS,CINREC,CPATH,CDATE,CTIME,CUNITS,
     . CTYPE,EVRESP,ID_DPL_ALT, BASE_YR
C
      INTEGER NULOC,IDENT(KULOC),NP(6,KULOC),IFLTAB(1200),NVRESP(6,6),
     . NC_ID_DPL_ALT, MX_LOC_DPL(2), IX_YEAR
      REAL VALUES(12),EVRATE(6),EVDIST(12),YR_EVRATE(6),
     . VAL_MON(12*KYEAR,KULOC), VAL_YR(KYEAR,KULOC)
C
      COMMON /AA/ NULOC,IDENT,NP,IFLTAB,VALUES,EVRATE,EVDIST,NVRESP,
     . NC_ID_DPL_ALT, MX_LOC_DPL, YR_EVRATE, VAL_MON, VAL_YR,
     . IX_YEAR
C     ----------------------------------------------------------------
```

RDMATF

Purpose of RDMATF

RDMATF reads the output file created by MRD's postprocessing program "V1.EXE" and stores the output data in an HECDSS data file. Any monthly regular interval time series data may be read by RDMATF as long as the data is stored in columns; e.g. All Ft. Peck release is in column 1, all Garrison release is in column 2 ....etc. and there is one month per line. The data is read free-format so that they need not be written in fixed locations.

Copying the Program onto Your Hard Disk

Use the "pkunzip.exe" utility to retrieve and uncompress the program from diskette. If "pkunzip.exe" does not already exist on your system, copy it from the diskette onto your fixed disk drive (hard disk). Normally, it would be copied into the "C:\HECEXE" subdirectory and the "path" command in your "C:\AUTOEXEC.BAT" file would contain a reference to this subdirectory. To retrieve the program from diskette in drive "a:" and store it on your fixed disk drive in subdirectory "c:\hecexe", enter the following command:

```
pkunzip a:rdmatf c:\hecexe
```

Copying Example Data onto Your Hard Disk

To copy the example input data file "RC9881.TXT" onto your fixed disk drive (hard disk), follow the preceding instructions for copying the program "RDMATF.EXE" but substitute "rc9881" in place of "rdmatf".

Preparing the Input data file.

Output from the "VI.EXE" program (or any other similar file) need not be modified before it is read by "RDMATF.EXE". However, RDMATF requires that the first 2 lines of the file contain data normally written to the "dmatfile" by "v1.exe". The first line contains three parameters in the following order: (1) the simulation identification (e.g. "D0411212") which is used for part F of the DSS pathname, (2) the starting year of simulation (assumed to start in March), and (3) the ending year of simulation (assumed to end in February). The second line contains codes identifying the type and location of data written to the "dmatfile". There is one code for each column of data. The code consists of two parts: (a) the parameter identifier which is one character, and (b) the location identifier which is an integer from 1 to 13 (e.g. Fort Peck is location 1). RDMATF automatically determines all pathname parts as described in the document "HEC-PRM Pathname Parts", dated December 1991.

Example Input Data File

The following is the example data file (RC9881.TXT) included with the program:

```
 D0411212          1898  1980
G1  G2  G3  G5  G6  N8  N9  N10  N11  N12  N13
290.0  820.0  895.0  808.0  894.0  1024.0  1052.0  1124.0  1435.0  2051.0  3363.0
494.0  1464.0  2088.0  2177.0  2290.0  2410.0  2397.0  2396.0  2443.0  2911.0  3911.0
563.0  1542.0  1335.0  1526.0  1638.0  2090.0  2225.0  2492.0  3304.0  4747.0  7740.0
632.0  1707.0  1498.0  1541.0  1653.0  1873.0  2044.0  2440.0  3365.0  3692.0  4557.0
880.0  2336.0  1990.0  2140.0  2199.0  2294.0  2319.0  2430.0  2632.0  2867.0  3548.0
721.0  1954.0  2354.0  2346.0  2397.0  2468.0  2519.0  2770.0  3105.0  3385.0  4162.0
620.0  1693.0  2454.0  2455.0  2510.0  2569.0  2628.0  2879.0  3404.0  3779.0  4847.0
543.0  1649.0  1838.0  2418.0  2493.0  2513.0  2579.0  2784.0  3270.0  3666.0  4416.0
663.0  1678.0  1601.0  2317.0  2396.0  2412.0  2458.0  2588.0  2953.0  3219.0  3748.0
683.0  1382.0  2048.0  1373.0  1450.0  1458.0  1572.0  1809.0  2408.0  2761.0  3421.0
794.0  1587.0  1439.0  1106.0  1182.0  1184.0  1275.0  1532.0  2147.0  2328.0  2695.0
677.0  1534.0  1080.0  1008.0  1103.0  1137.0  1166.0  1221.0  1514.0  1596.0  1797.0
498.0  1310.0  1086.0  1400.0  1526.0  1697.0  1663.0  1682.0  1850.0  2428.0  3755.0
605.0  1723.0  1797.0  1911.0  2020.0  2148.0  2195.0  2290.0  2664.0  2843.0  3180.0
605.0  1919.0  1399.0  1490.0  1597.0  1952.0  2249.0  3115.0  5119.0  5781.0  7134.0
178.0  2003.0  1588.0  1724.0  1832.0  2100.0  2423.0  3481.0  5642.0  5916.0  6701.0
706.0  3258.0  2959.0  2897.0  2954.0  3019.0  3298.0  4424.0  6515.0  6650.0  7180.0
922.0  2336.0  3291.0  3381.0  3430.0  3482.0  3661.0  4403.0  5739.0  5808.0  6168.0
891.0  2107.0  3250.0  3285.0  3338.0  3376.0  3434.0  3657.0  4134.0  4392.0  5194.0
833.0  2003.0  2643.0  3228.0  3301.0  3394.0  3431.0  3541.0  3844.0  4045.0  4385.0
892.0  2054.0  2850.0  3532.0  3604.0  3680.0  3709.0  3806.0  4099.0  4225.0  4478.0
755.0  1399.0  2209.0  1494.0  1569.0  1619.0  1777.0  2028.0  2685.0  2935.0  3284.0
849.0  1613.0  1500.0  1106.0  1186.0  1208.0  1287.0  1485.0  1989.0  2114.0  2350.0
726.0  1587.0  1082.0  1008.0  1109.0  1257.0  1263.0  1278.0  1444.0  1490.0  1619.0
351.0  991.0  1162.0  1151.0  1286.0  1839.0  1875.0  2030.0  2636.0  3358.0  4967.0
351.0  1176.0  1463.0  1623.0  1740.0  2090.0  2165.0  2332.0  2939.0  3084.0  3367.0
418.0  1243.0  1889.0  1900.0  2014.0  2073.0  2177.0  2401.0  3073.0  3193.0  3428.0
553.0  1429.0  1730.0  1832.0  1946.0  2019.0  2110.0  2314.0  2754.0  2810.0  3057.0
586.0  1571.0  1501.0  1674.0  1736.0  2023.0  2142.0  2471.0  3115.0  3205.0  3560.0
554.0  1429.0  1691.0  1694.0  1747.0  1938.0  2053.0  2495.0  3266.0  3328.0  3638.0
462.0  1175.0  1643.0  1672.0  1729.0  1879.0  1992.0  2401.0  3237.0  3294.0  3688.0
378.0  1069.0  1096.0  1858.0  1936.0  2009.0  2100.0  2425.0  3138.0  3474.0  4087.0
276.0  800.0  1127.0  1636.0  1718.0  1783.0  1920.0  2304.0  3174.0  3528.0  4207.0
721.0  1468.0  1635.0  922.0  1001.0  1043.0  1125.0  1323.0  1831.0  2121.0  2697.0
```

{additional data deleted}

```
758.0  1273.0  1695.0  2313.0  2431.0  2762.0  3155.0  3415.0  4136.0  4199.0  4399.0
695.0  1395.0  1978.0  1358.0  1512.0  1757.0  2023.0  2316.0  2893.0  2918.0  3389.0
812.0  1605.0  1500.0  1106.0  1204.0  1335.0  1563.0  1716.0  2142.0  2178.0  2492.0
834.0  1595.0  1004.0  1008.0  1163.0  1321.0  1472.0  1723.0  2061.0  2171.0  2769.0
396.0  1067.0  1364.0  1243.0  1425.0  1630.0  1754.0  2273.0  3177.0  3293.0  4397.0
445.0  1309.0  1457.0  1493.0  1697.0  1866.0  2018.0  2566.0  4875.0  5649.0  7146.0
529.0  1379.0  1597.0  1610.0  1775.0  1925.0  2038.0  2594.0  2949.0  3088.0  3359.0
608.0  1490.0  1556.0  1563.0  1702.0  1865.0  2139.0  2855.0  3924.0  4169.0  4745.0
792.0  1602.0  2359.0  2266.0  2298.0  2387.0  2374.0  2415.0  2554.0  2601.0  2893.0
865.0  1538.0  2152.0  2170.0  2242.0  2383.0  2415.0  2467.0  2627.0  2908.0  2987.0
826.0  1313.0  2209.0  2147.0  2196.0  2314.0  2357.0  2364.0  2537.0  2865.0  2969.0
730.0  1274.0  1442.0  2210.0  2257.0  2405.0  2479.0  2540.0  2611.0  2674.0  2702.0
538.0  966.0  1460.0  1920.0  1993.0  2177.0  2226.0  2359.0  2446.0  2627.0  2697.0
780.0  1538.0  1524.0  816.0  916.0  1041.0  1124.0  1266.0  1638.0  2199.0  2414.0
942.0  1745.0  1126.0  816.0  932.0  1029.0  1055.0  1224.0  1416.0  1379.0  1489.0
884.0  1722.0  836.0  744.0  856.0  921.0  989.0  1124.0  1287.0  1347.0  1549.0
```

Executing the program.

To execute the program, you must define the input data file and the output HECDSS data file on the execution line. The "MENUPRM" program will assist you in defining these files. It assumes that all "dmatfile"s have the extension ".mat". To execute the program from the DOS prompt, the program name and file assignments must be entered. For example, to use the above example file named "RC9881.TXT" and store it an HECDSS data file named "EXAMPLE.DSS", enter the following command at the DOS prompt:

```
RDMATF I=RC9881.TXT DSS=EXAMPLE
```

Example Output Data File

```
     -----DSS---ZOPEN:  Existing File Opened,  File: RDMATF.DSS
                        Unit:  71;  DSS Version: 6-GO
-----DSS---ZWRITE Unit  71; Vers.    1:  //FTPK-GARR/FLOW/01JAN1890/1MON/D0411212/
-----DSS---ZWRITE Unit  71; Vers.    1:  //FTPK-GARR/FLOW/01JAN1900/1MON/D0411212/
-----DSS---ZWRITE Unit  71; Vers.    1:  //FTPK-GARR/FLOW/01JAN1910/1MON/D0411212/
-----DSS---ZWRITE Unit  71; Vers.    1:  //FTPK-GARR/FLOW/01JAN1920/1MON/D0411212/
-----DSS---ZWRITE Unit  71; Vers.    1:  //FTPK-GARR/FLOW/01JAN1930/1MON/D0411212/
-----DSS---ZWRITE Unit  71; Vers.    1:  //FTPK-GARR/FLOW/01JAN1940/1MON/D0411212/
-----DSS---ZWRITE Unit  71; Vers.    1:  //FTPK-GARR/FLOW/01JAN1950/1MON/D0411212/
-----DSS---ZWRITE Unit  71; Vers.    1:  //FTPK-GARR/FLOW/01JAN1960/1MON/D0411212/
-----DSS---ZWRITE Unit  71; Vers.    1:  //FTPK-GARR/FLOW/01JAN1970/1MON/D0411212/
-----DSS---ZWRITE Unit  71; Vers.    1:  //FTPK-GARR/FLOW/01JAN1980/1MON/D0411212/
-----DSS---ZWRITE Unit  71; Vers.    1:  //GARR-OAHE/FLOW/01JAN1890/1MON/D0411212/
-----DSS---ZWRITE Unit  71; Vers.    1:  //GARR-OAHE/FLOW/01JAN1900/1MON/D0411212/
-----DSS---ZWRITE Unit  71; Vers.    1:  //GARR-OAHE/FLOW/01JAN1910/1MON/D0411212/
-----DSS---ZWRITE Unit  71; Vers.    1:  //GARR-OAHE/FLOW/01JAN1920/1MON/D0411212/
-----DSS---ZWRITE Unit  71; Vers.    1:  //GARR-OAHE/FLOW/01JAN1930/1MON/D0411212/
-----DSS---ZWRITE Unit  71; Vers.    1:  //GARR-OAHE/FLOW/01JAN1940/1MON/D0411212/
-----DSS---ZWRITE Unit  71; Vers.    1:  //GARR-OAHE/FLOW/01JAN1950/1MON/D0411212/
-----DSS---ZWRITE Unit  71; Vers.    1:  //GARR-OAHE/FLOW/01JAN1960/1MON/D0411212/
-----DSS---ZWRITE Unit  71; Vers.    1:  //GARR-OAHE/FLOW/01JAN1970/1MON/D0411212/
-----DSS---ZWRITE Unit  71; Vers.    1:  //GARR-OAHE/FLOW/01JAN1980/1MON/D0411212/
-----DSS---ZWRITE Unit  71; Vers.    1:  //OAHE-BEND/FLOW/01JAN1890/1MON/D0411212/
-----DSS---ZWRITE Unit  71; Vers.    1:  //OAHE-BEND/FLOW/01JAN1900/1MON/D0411212/
-----DSS---ZWRITE Unit  71; Vers.    1:  //OAHE-BEND/FLOW/01JAN1910/1MON/D0411212/
-----DSS---ZWRITE Unit  71; Vers.    1:  //OAHE-BEND/FLOW/01JAN1920/1MON/D0411212/
-----DSS---ZWRITE Unit  71; Vers.    1:  //OAHE-BEND/FLOW/01JAN1930/1MON/D0411212/
-----DSS---ZWRITE Unit  71; Vers.    1:  //OAHE-BEND/FLOW/01JAN1940/1MON/D0411212/
-----DSS---ZWRITE Unit  71; Vers.    1:  //OAHE-BEND/FLOW/01JAN1950/1MON/D0411212/
-----DSS---ZWRITE Unit  71; Vers.    1:  //OAHE-BEND/FLOW/01JAN1960/1MON/D0411212/
-----DSS---ZWRITE Unit  71; Vers.    1:  //OAHE-BEND/FLOW/01JAN1970/1MON/D0411212/
-----DSS---ZWRITE Unit  71; Vers.    1:  //OAHE-BEND/FLOW/01JAN1980/1MON/D0411212/
-----DSS---ZWRITE Unit  71; Vers.    1:  //FTRA-GAPT/FLOW/01JAN1890/1MON/D0411212/
-----DSS---ZWRITE Unit  71; Vers.    1:  //FTRA-GAPT/FLOW/01JAN1900/1MON/D0411212/
-----DSS---ZWRITE Unit  71; Vers.    1:  //FTRA-GAPT/FLOW/01JAN1910/1MON/D0411212/
-----DSS---ZWRITE Unit  71; Vers.    1:  //FTRA-GAPT/FLOW/01JAN1920/1MON/D0411212/
-----DSS---ZWRITE Unit  71; Vers.    1:  //FTRA-GAPT/FLOW/01JAN1930/1MON/D0411212/
-----DSS---ZWRITE Unit  71; Vers.    1:  //FTRA-GAPT/FLOW/01JAN1940/1MON/D0411212/
-----DSS---ZWRITE Unit  71; Vers.    1:  //FTRA-GAPT/FLOW/01JAN1950/1MON/D0411212/
-----DSS---ZWRITE Unit  71; Vers.    1:  //FTRA-GAPT/FLOW/01JAN1960/1MON/D0411212/
-----DSS---ZWRITE Unit  71; Vers.    1:  //FTRA-GAPT/FLOW/01JAN1970/1MON/D0411212/
-----DSS---ZWRITE Unit  71; Vers.    1:  //FTRA-GAPT/FLOW/01JAN1980/1MON/D0411212/
-----DSS---ZWRITE Unit  71; Vers.    1:  //GAPT-SUX/FLOW/01JAN1890/1MON/D0411212/
-----DSS---ZWRITE Unit  71; Vers.    1:  //GAPT-SUX/FLOW/01JAN1900/1MON/D0411212/
-----DSS---ZWRITE Unit  71; Vers.    1:  //GAPT-SUX/FLOW/01JAN1910/1MON/D0411212/
-----DSS---ZWRITE Unit  71; Vers.    1:  //GAPT-SUX/FLOW/01JAN1920/1MON/D0411212/
-----DSS---ZWRITE Unit  71; Vers.    1:  //GAPT-SUX/FLOW/01JAN1930/1MON/D0411212/
-----DSS---ZWRITE Unit  71; Vers.    1:  //GAPT-SUX/FLOW/01JAN1940/1MON/D0411212/
-----DSS---ZWRITE Unit  71; Vers.    1:  //GAPT-SUX/FLOW/01JAN1950/1MON/D0411212/
-----DSS---ZWRITE Unit  71; Vers.    1:  //GAPT-SUX/FLOW/01JAN1960/1MON/D0411212/
-----DSS---ZWRITE Unit  71; Vers.    1:  //GAPT-SUX/FLOW/01JAN1970/1MON/D0411212/
-----DSS---ZWRITE Unit  71; Vers.    1:  //GAPT-SUX/FLOW/01JAN1980/1MON/D0411212/
-----DSS---ZWRITE Unit  71; Vers.    1:  //SUX-OMA/FLOW/01JAN1890/1MON/D0411212/
-----DSS---ZWRITE Unit  71; Vers.    1:  //SUX-OMA/FLOW/01JAN1900/1MON/D0411212/
-----DSS---ZWRITE Unit  71; Vers.    1:  //SUX-OMA/FLOW/01JAN1910/1MON/D0411212/
-----DSS---ZWRITE Unit  71; Vers.    1:  //SUX-OMA/FLOW/01JAN1920/1MON/D0411212/
-----DSS---ZWRITE Unit  71; Vers.    1:  //SUX-OMA/FLOW/01JAN1930/1MON/D0411212/
-----DSS---ZWRITE Unit  71; Vers.    1:  //SUX-OMA/FLOW/01JAN1940/1MON/D0411212/
-----DSS---ZWRITE Unit  71; Vers.    1:  //SUX-OMA/FLOW/01JAN1950/1MON/D0411212/
-----DSS---ZWRITE Unit  71; Vers.    1:  //SUX-OMA/FLOW/01JAN1960/1MON/D0411212/
-----DSS---ZWRITE Unit  71; Vers.    1:  //SUX-OMA/FLOW/01JAN1970/1MON/D0411212/
-----DSS---ZWRITE Unit  71; Vers.    1:  //SUX-OMA/FLOW/01JAN1980/1MON/D0411212/
-----DSS---ZWRITE Unit  71; Vers.    1:  //OMA-NCNE/FLOW/01JAN1890/1MON/D0411212/
-----DSS---ZWRITE Unit  71; Vers.    1:  //OMA-NCNE/FLOW/01JAN1900/1MON/D0411212/
-----DSS---ZWRITE Unit  71; Vers.    1:  //OMA-NCNE/FLOW/01JAN1910/1MON/D0411212/
-----DSS---ZWRITE Unit  71; Vers.    1:  //OMA-NCNE/FLOW/01JAN1920/1MON/D0411212/
-----DSS---ZWRITE Unit  71; Vers.    1:  //OMA-NCNE/FLOW/01JAN1930/1MON/D0411212/
-----DSS---ZWRITE Unit  71; Vers.    1:  //OMA-NCNE/FLOW/01JAN1940/1MON/D0411212/
-----DSS---ZWRITE Unit  71; Vers.    1:  //OMA-NCNE/FLOW/01JAN1950/1MON/D0411212/
-----DSS---ZWRITE Unit  71; Vers.    1:  //OMA-NCNE/FLOW/01JAN1960/1MON/D0411212/
-----DSS---ZWRITE Unit  71; Vers.    1:  //OMA-NCNE/FLOW/01JAN1970/1MON/D0411212/
-----DSS---ZWRITE Unit  71; Vers.    1:  //OMA-NCNE/FLOW/01JAN1980/1MON/D0411212/
-----DSS---ZWRITE Unit  71; Vers.    1:  //NCNE-MKC/FLOW/01JAN1890/1MON/D0411212/
-----DSS---ZWRITE Unit  71; Vers.    1:  //NCNE-MKC/FLOW/01JAN1900/1MON/D0411212/
-----DSS---ZWRITE Unit  71; Vers.    1:  //NCNE-MKC/FLOW/01JAN1910/1MON/D0411212/
-----DSS---ZWRITE Unit  71; Vers.    1:  //NCNE-MKC/FLOW/01JAN1920/1MON/D0411212/
-----DSS---ZWRITE Unit  71; Vers.    1:  //NCNE-MKC/FLOW/01JAN1930/1MON/D0411212/
-----DSS---ZWRITE Unit  71; Vers.    1:  //NCNE-MKC/FLOW/01JAN1940/1MON/D0411212/
-----DSS---ZWRITE Unit  71; Vers.    1:  //NCNE-MKC/FLOW/01JAN1950/1MON/D0411212/
-----DSS---ZWRITE Unit  71; Vers.    1:  //NCNE-MKC/FLOW/01JAN1960/1MON/D0411212/
-----DSS---ZWRITE Unit  71; Vers.    1:  //NCNE-MKC/FLOW/01JAN1970/1MON/D0411212/
-----DSS---ZWRITE Unit  71; Vers.    1:  //NCNE-MKC/FLOW/01JAN1980/1MON/D0411212/
-----DSS---ZWRITE Unit  71; Vers.    1:  //MKC-BNMO/FLOW/01JAN1890/1MON/D0411212/
-----DSS---ZWRITE Unit  71; Vers.    1:  //MKC-BNMO/FLOW/01JAN1900/1MON/D0411212/
-----DSS---ZWRITE Unit  71; Vers.    1:  //MKC-BNMO/FLOW/01JAN1910/1MON/D0411212/
-----DSS---ZWRITE Unit  71; Vers.    1:  //MKC-BNMO/FLOW/01JAN1920/1MON/D0411212/
-----DSS---ZWRITE Unit  71; Vers.    1:  //MKC-BNMO/FLOW/01JAN1930/1MON/D0411212/
```

```
-----DSS---ZWRITE Unit  71; Vers.    1:  //MKC-BNMO/FLOW/01JAN1940/1MON/D0411212/
-----DSS---ZWRITE Unit  71; Vers.    1:  //MKC-BNMO/FLOW/01JAN1950/1MON/D0411212/
-----DSS---ZWRITE Unit  71; Vers.    1:  //MKC-BNMO/FLOW/01JAN1960/1MON/D0411212/
-----DSS---ZWRITE Unit  71; Vers.    1:  //MKC-BNMO/FLOW/01JAN1970/1MON/D0411212/
-----DSS---ZWRITE Unit  71; Vers.    1:  //MKC-BNMO/FLOW/01JAN1980/1MON/D0411212/
-----DSS---ZWRITE Unit  71; Vers.    1:  //BNMO-HEMO/FLOW/01JAN1890/1MON/D0411212/
-----DSS---ZWRITE Unit  71; Vers.    1:  //BNMO-HEMO/FLOW/01JAN1900/1MON/D0411212/
-----DSS---ZWRITE Unit  71; Vers.    1:  //BNMO-HEMO/FLOW/01JAN1910/1MON/D0411212/
-----DSS---ZWRITE Unit  71; Vers.    1:  //BNMO-HEMO/FLOW/01JAN1920/1MON/D0411212/
-----DSS---ZWRITE Unit  71; Vers.    1:  //BNMO-HEMO/FLOW/01JAN1930/1MON/D0411212/
-----DSS---ZWRITE Unit  71; Vers.    1:  //BNMO-HEMO/FLOW/01JAN1940/1MON/D0411212/
-----DSS---ZWRITE Unit  71; Vers.    1:  //BNMO-HEMO/FLOW/01JAN1950/1MON/D0411212/
-----DSS---ZWRITE Unit  71; Vers.    1:  //BNMO-HEMO/FLOW/01JAN1960/1MON/D0411212/
-----DSS---ZWRITE Unit  71; Vers.    1:  //BNMO-HEMO/FLOW/01JAN1970/1MON/D0411212/
-----DSS---ZWRITE Unit  71; Vers.    1:  //BNMO-HEMO/FLOW/01JAN1980/1MON/D0411212/
-----DSS---ZWRITE Unit  71; Vers.    1:  //HEMO-STL/FLOW/01JAN1890/1MON/D0411212/
-----DSS---ZWRITE Unit  71; Vers.    1:  //HEMO-STL/FLOW/01JAN1900/1MON/D0411212/
-----DSS---ZWRITE Unit  71; Vers.    1:  //HEMO-STL/FLOW/01JAN1910/1MON/D0411212/
-----DSS---ZWRITE Unit  71; Vers.    1:  //HEMO-STL/FLOW/01JAN1920/1MON/D0411212/
-----DSS---ZWRITE Unit  71; Vers.    1:  //HEMO-STL/FLOW/01JAN1930/1MON/D0411212/
-----DSS---ZWRITE Unit  71; Vers.    1:  //HEMO-STL/FLOW/01JAN1940/1MON/D0411212/
-----DSS---ZWRITE Unit  71; Vers.    1:  //HEMO-STL/FLOW/01JAN1950/1MON/D0411212/
-----DSS---ZWRITE Unit  71; Vers.    1:  //HEMO-STL/FLOW/01JAN1960/1MON/D0411212/
-----DSS---ZWRITE Unit  71; Vers.    1:  //HEMO-STL/FLOW/01JAN1970/1MON/D0411212/
-----DSS---ZWRITE Unit  71; Vers.    1:  //HEMO-STL/FLOW/01JAN1980/1MON/D0411212/
   -----DSS---ZCLOSE Unit:  71,   File: RDMATF.DSS
              Pointer Utilization:  0.30
              Number of Records:    200
              File Size:    227.1  Kbytes
              Percent Inactive:   0.0
```

Listing of Computer Source Code for RDMATF

```
      PROGRAM RDMATF
C
C     Read MRD DMATFILE and store results in DSS data file
C
      PARAMETER (KVAR=30,KSDATA=20000)
      INTEGER*4 JULWIN(2),INTRVL
C      INTEGER JULWIN(2),INTRVL
      INTEGER YRWIN(2),MONWIN(2),DAYWIN(2),MINWIN(2),NPATH(KVAR),
     .IFLTAB(1200), NPNPRT(6), IX_PARAM(KVAR), IX_LOC(KVAR)
C
      CHARACTER CFNIO*64,CFNDSS*64,CPATH(KVAR)*80,
     .CNTRVL*9,DATWIN(2)*9,HOUWIN(2)*4,CUNITS*8,CTYPE*8,CPNPRT(6)*32,
     .USR_VAR(KVAR)*4
C
      REAL TSDATA(KSDATA)
C
      include 'io.cbl'
      include 'param.cbl'
C
C
      CALL ATTACH(5,'INPUT','STDIN',' ',CFNIO,IOER01)
      CALL ATTACH(6,'OUTPUT','STDOUT',' ',CFNIO,IOER02)
      CALL ATTACH(21,'DMATFILE','DMATFILE.MAT',' ',CFNIO,IOER03)
      CALL ATTACH(71,'DSS_FILE','SCRATCH31','NOP',CFNDSS,IOERDS)
      CALL ATTEND
C
      CDELM=',/'
      CALL SETDLM(1,CDELM,1,2,ITBL)
      CALL SETDLM(2,CDELM,1,-1,ITBL)
      CALL SETDLM(3,CDELM,1,-1,ITBL)
C
      NVAR=0
      MINWIN(1)=1440
      MINWIN(2)=1440
      CPNPRT(4)=' '
C
 1500 JREC=1
      DO WHILE(JREC.LE.2)
        READ(21,1505,END=9000) CINREC
        CAPINP=CINREC
        CALL UPCASE(CAPINP)
 1505   FORMAT(A)
C
        CALL FINDLM (CINREC,1,KINREC,NFL,IBF,LNF,IDT,IDP,ITBL)
C
        IF(JREC.EQ.1) THEN
          JFL=1
          DO WHILE(JFL.LE.NFL)
            IF(JFL.EQ.1) THEN
              CPNPRT(6)=CINREC(IBF(JFL):IBF(JFL)+LNF(JFL)-1)
              CALL CHRLNB(CPNPRT(6),NPNPRT(6))
C
            ELSE IF(JFL.EQ.2) THEN
              MONWIN(1)=3
              DAYWIN(1)=31
              ITP=INTGR(CINREC,IBF(JFL),LNF(JFL),IER01)
              IF(IER01.EQ.0) THEN
                YRWIN(1)=ITP
              ELSE
                YRWIN(1)=1898
              ENDIF
C
            ELSE IF(JFL.EQ.3) THEN
              MONWIN(2)=2
              DAYWIN(2)=28
              ITP=INTGR(CINREC,IBF(JFL),LNF(JFL),IER01)
              IF(IER01.EQ.0) THEN
                YRWIN(2)=ITP
              ELSE
                YRWIN(2)=1990
              ENDIF
              IF(MOD(YRWIN(2),4).EQ.0 .AND. MOD(YRWIN(2),100).NE.0
     .        .AND. MOD(YRWIN(2),400).NE.0) DAYWIN(2)=29
C
            ENDIF
            JFL=JFL+1
          END DO
C
        ELSE IF(JREC.EQ.2) THEN
C
          JFL=1
          DO WHILE (JFL.LE.NFL)
            IF(LNF(JFL).LT.1) THEN
            ELSE
              NVAR=NVAR+1
              USR_VAR(NVAR)=CAPINP(IBF(JFL):IBF(JFL)+LNF(JFL)-1)
              IX_PARAM(NVAR)=ICHAR(USR_VAR(NVAR)(1:1))-64
              CALL CHRRJS(CAPINP(IBF(JFL)+1:IBF(JFL)+LNF(JFL)-1),
```

```
     .        CTEMP(1:5))
              READ(CTEMP,'(I5)') IX_LOC(NVAR)
            ENDIF
            JFL=JFL+1
          END DO
        ENDIF
        JREC=JREC+1
      END DO
C
      J=1
      DO WHILE(J.LE.2)
        JULWIN(J)=IYMDJL(YRWIN(J),MONWIN(J),DAYWIN(J))
        J=J+1
      END DO
C
C     Determine the number of time periods
C     ------------------------------------------------------------
 2000 IFLAG1=1
      CALL ZGINTL (INTRVL,'1MON',NTEMP,IFLAG1)
      IFLAG2=0
      NPER=1+NOPERS(INTRVL,IFLAG2,JULWIN(1),MINWIN(1),JULWIN(2),
     .MINWIN(2))
C
      J=1
      DO WHILE (J.LE.2)
        DATWIN(J)=' '
        HOUWIN(J)=' '
        CALL JULDAT (JULWIN(J),104,DATWIN(J),NTEMP)
        ITP=M2IHM (MINWIN(J),HOUWIN(J))
        J=J+1
      END DO
      CNTRVL='1MON'
C
      J=1
      DO WHILE(J.LE.NPER*NVAR)
        TSDATA(J)=-901.0E+30
        J=J+1
      END DO
C
C
C     Read data
C     ------------------------------------------------------------
      IORD=1
      DO WHILE(IORD.LE.NPER)
        READ(21,'(A)',END=3500) CINREC
        CALL FINDLM (CINREC,1,KINREC,NFL,IBF,LNF,IDT,IDP,ITBL)
        JFL=1
        DO WHILE(JFL.LE.NFL)
          JTP=(JFL-1)*NPER+IORD
          TEMP=XREAL(CINREC,IBF(JFL),LNF(JFL),IER02)
          IF(IER02.EQ.0) TSDATA(JTP)=TEMP
          JFL=JFL+1
        END DO
        IORD=IORD+1
      END DO
C
C
C     Store data in DSS file
C
 3500 CALL ZOPEN(IFLTAB,CFNDSS,IODSS)
      JVAR=1
      DO WHILE (JVAR.LE.NVAR)
        CPNPRT(1)=ID_RIV_NAME(IX_LOC(JVAR))
        CALL CHRLNB(CPNPRT(1),NPNPRT(1))
C
        JX_PARAM=IX_PARAM(JVAR)
        IF(INDEX(ID_PARAM_NAME(JX_PARAM),'RELEASE').EQ.1 .OR.
        INDEX(ID_PARAM_NAME(JX_PARAM),'FLOW').EQ.1) THEN
          CPNPRT(2)=ID_LOC_NAME_FLOW(IX_LOC(JVAR))
        ELSE
          CPNPRT(2)=ID_LOC_NAME(IX_LOC(JVAR))
        ENDIF
        CALL CHRLNB(CPNPRT(2),NPNPRT(2))
C
        CPNPRT(3)=ID_PARAM_NAME(IX_PARAM(JVAR))
        CPNPRT(3)=ID_PARAM_PNPRT(IX_PARAM(JVAR))
        CALL CHRLNB(CPNPRT(3),NPNPRT(3))
        CPNPRT(5)=CNTRVL
        CALL CHRLNB(CPNPRT(5),NPNPRT(5))
        CALL ZPATH(CPNPRT(1),CPNPRT(2),CPNPRT(3),CPNPRT(4),CPNPRT(5),
        CPNPRT(6),CPATH(JVAR),NPATH(JVAR))
C
        JX_PARAM=IX_PARAM(JVAR)
        IF(INDEX(ID_PARAM_NAME(JX_PARAM),'KAF').GT.0) THEN
          CUNITS='KAF'
        ELSE IF(INDEX(ID_PARAM_NAME(JX_PARAM),'1000AF').GT.0) THEN
          CUNITS='KAF'
        ELSE IF(INDEX(ID_PARAM_NAME(JX_PARAM),'1000 AF').GT.0) THEN
          CUNITS='KAF'
        ELSE IF(INDEX(ID_PARAM_NAME(JX_PARAM),'FT').GT.0) THEN
          CUNITS='FT'
```

```
          ELSE IF(INDEX(ID_PARAM_NAME(JX_PARAM),'KCFS').GT.0) THEN
            CUNITS='KCFS'
          ELSE IF(INDEX(ID_PARAM_NAME(JX_PARAM),'1000CFS').GT.0) THEN
            CUNITS='KCFS'
          ELSE IF(INDEX(ID_PARAM_NAME(JX_PARAM),'KMWH').GT.0) THEN
            CUNITS='KMWH'
          ELSE IF(INDEX(ID_PARAM_NAME(JX_PARAM),'MW').GT.0) THEN
            CUNITS='MW'
          ELSE IF(INDEX(ID_PARAM_NAME(JX_PARAM),'LENGTH MO').GT.0) THEN
            CUNITS='MONTHS'
          ELSE
            CUNITS=' '
          ENDIF
C
          IF(INDEX(ID_PARAM_NAME(JVAR),'STORAGE').GT.0) THEN
            CTYPE='INST-VAL'
          ELSE IF(INDEX(ID_PARAM_NAME(JVAR),'POOL ELEV').GT.0) THEN
            CTYPE='INST-VAL'
          ELSE IF(INDEX(ID_PARAM_NAME(JVAR),'PEAK POWER').GT.0) THEN
            CTYPE='INST-VAL'
          ELSE IF(INDEX(ID_PARAM_NAME(JVAR),'ENERGY').GT.0) THEN
            CTYPE='PER-CUM '
          ELSE IF(INDEX(ID_PARAM_NAME(JVAR),'NAV LENGTH').GT.0) THEN
            CTYPE='INST-VAL'
          ELSE
            CTYPE='PER-AVER'
          ENDIF
C
          IPLAN=0
          ISDATA=1+(JVAR-1)*NPER
          CALL ZSRTS (IFLTAB,CPATH(JVAR)(1:NPATH(JVAR)),DATWIN(1),'2400',
     .    NPER,TSDATA(ISDATA),CUNITS,CTYPE,IPLAN,IER03)
C
          JVAR=JVAR+1
        END DO
        CALL ZCLOSE(IFLTAB)
C
C       It's all over folks!
C       ------------------------------------------------------------------
 9000   STOP
        END
```

```
      BLOCK DATA
C
      INCLUDE 'IO.CBL'
      include 'param.cbl'
C
      DATA LERROR/.FALSE./
C
      DATA KDI,KDO,KUTR /5,6,29/
      DATA LECHO/.TRUE./
      DATA CDELM/',-   '/
      DATA CIDMON /'JAN','FEB','MAR','APR','MAY','JUN','JUL',
     .'AUG','SEP','OCT','NOV','DEC'/
      DATA DAYMON/31,28,31,30,31,30,31,31,30,31,30,31/
C
C
      DATA ID_PARAM_NAME( 1)/'REACH INF,KAF   '/
      DATA ID_PARAM_NAME( 2)/'EVAP,1000 AF    '/
      DATA ID_PARAM_NAME( 3)/'INF ADJST,KAF   '/
      DATA ID_PARAM_NAME( 4)/'MOD INF,KAF     '/
      DATA ID_PARAM_NAME( 5)/'STORAGE,KAF     '/
      DATA ID_PARAM_NAME( 6)/'POOL ELEV,FT    '/
      DATA ID_PARAM_NAME( 7)/'RELEASE,KAF     '/
      DATA ID_PARAM_NAME( 8)/'RELEASE,KCFS    '/
      DATA ID_PARAM_NAME( 9)/'AVE POWER,MW    '/
      DATA ID_PARAM_NAME(10)/'PEAK POWER,MW   '/
      DATA ID_PARAM_NAME(11)/'ENERGY,KMWH     '/
      DATA ID_PARAM_NAME(12)/'NAV LVL,KCFS    '/
      DATA ID_PARAM_NAME(13)/'NAV LENGTH MO   '/
      DATA ID_PARAM_NAME(14)/'FLOW, 1000AF    '/
      DATA ID_PARAM_NAME(15)/'FLOW, 1000CFS   '/
C
C
      DATA ID_PARAM_PNPRT( 1)/'FLOW_LOC_INC            '/
      DATA ID_PARAM_PNPRT( 2)/'EVAP                    '/
      DATA ID_PARAM_PNPRT( 3)/'FLOW_LOC_DPL            '/
      DATA ID_PARAM_PNPRT( 4)/'FLOW_LOC                '/
      DATA ID_PARAM_PNPRT( 5)/'STOR                    '/
      DATA ID_PARAM_PNPRT( 6)/'ELEV                    '/
      DATA ID_PARAM_PNPRT( 7)/'FLOW                    '/
      DATA ID_PARAM_PNPRT( 8)/'FLOW(KCFS)              '/
      DATA ID_PARAM_PNPRT( 9)/'AVE POWER,MW            '/
      DATA ID_PARAM_PNPRT(10)/'PEAK POWER,MW           '/
      DATA ID_PARAM_PNPRT(11)/'ENERGY                  '/
      DATA ID_PARAM_PNPRT(12)/'FLOW_NAV_LVL(KCFS)      '/
      DATA ID_PARAM_PNPRT(13)/'MON_NAV_LENGTH          '/
      DATA ID_PARAM_PNPRT(14)/'FLOW                    '/
      DATA ID_PARAM_PNPRT(15)/'FLOW(KCFS)              '/
C
      DATA ID_LOC_NAME( 1)/'FTPK  '/,
     .                         ID_RIV_NAME( 1)/'                    '/
      DATA ID_LOC_NAME( 2)/'GARR  '/,
     .                         ID_RIV_NAME( 2)/'                    '/
      DATA ID_LOC_NAME( 3)/'OAHE  '/,
     .                         ID_RIV_NAME( 3)/'                    '/
      DATA ID_LOC_NAME( 4)/'BEND  '/,
     .                         ID_RIV_NAME( 4)/'                    '/
      DATA ID_LOC_NAME( 5)/'FTRA  '/,
     .                         ID_RIV_NAME( 5)/'                    '/
      DATA ID_LOC_NAME( 6)/'GAPT  '/,
     .                         ID_RIV_NAME( 6)/'                    '/
      DATA ID_LOC_NAME( 7)/'SYS   '/,
     .                         ID_RIV_NAME( 7)/'                    '/
      DATA ID_LOC_NAME( 8)/'SUX   '/,
     .                         ID_RIV_NAME( 8)/'                    '/
      DATA ID_LOC_NAME( 9)/'OMA   '/,
     .                         ID_RIV_NAME( 9)/'                    '/
      DATA ID_LOC_NAME(10)/'NCNE  '/,
     .                         ID_RIV_NAME(10)/'                    '/
      DATA ID_LOC_NAME(11)/'MKC   '/,
     .                         ID_RIV_NAME(11)/'                    '/
      DATA ID_LOC_NAME(12)/'BNMO  '/,
     .                         ID_RIV_NAME(12)/'                    '/
      DATA ID_LOC_NAME(13)/'HEMO  '/,
     .                         ID_RIV_NAME(13)/'                    '/
C
      DATA ID_LOC_NAME_FLOW( 1)/'FTPK-GARR'/
      DATA ID_LOC_NAME_FLOW( 2)/'GARR-OAHE'/
      DATA ID_LOC_NAME_FLOW( 3)/'OAHE-BEND'/
      DATA ID_LOC_NAME_FLOW( 4)/'BEND-FTRA'/
      DATA ID_LOC_NAME_FLOW( 5)/'FTRA-GAPT'/
      DATA ID_LOC_NAME_FLOW( 6)/'GAPT-SUX '/
      DATA ID_LOC_NAME_FLOW( 7)/'SYS      '/
      DATA ID_LOC_NAME_FLOW( 8)/'SUX-OMA  '/
      DATA ID_LOC_NAME_FLOW( 9)/'OMA-NCNE '/
      DATA ID_LOC_NAME_FLOW(10)/'NCNE-MKC '/
      DATA ID_LOC_NAME_FLOW(11)/'MKC-BNMO '/
      DATA ID_LOC_NAME_FLOW(12)/'BNMO-HEMO'/
      DATA ID_LOC_NAME_FLOW(13)/'HEMO-STL '/
C
      END
```

```
C                                                              BLOCK IO
      PARAMETER (KINREC=132)
      CHARACTER CINREC*(KINREC),BINREC*(KINREC),CAPINP*(KINREC),
     . CDELM*5,CSYSDA*9,CSYSTM*8,CPROGN*17,CIDMON(12)*3,CTEMP*132,
     . EJECTP*1
      COMMON /IOC/ CINREC,BINREC,CAPINP,CDELM,CSYSDA,CSYSTM,CPROGN,
     . CIDMON,CTEMP,EJECTP
C
      PARAMETER (KFL=30)
      LOGICAL LBATCH,LERROR,LSTOP,LEOF,LECHO,LNEWBA,LBLANK,LCOMND,
     .LDTK14,LDCALC,LVERIF,LPRMPT,LEXIT,LFINIS,LNEXT,LALL,LCCPL,LHPPL,
     .LVERPL,LZETPL,LOPENP
C
      INTEGER NFL,IBF(KFL),IEF(KFL),LNF(KFL),IDT(KFL),IDP(KFL),
     .ITBL(128),KDI,KDO,KUTR,KDNME,KDDES,KDTR,ISYSMN,NPROGN,DAYMON(12)
C
      INTEGER MSYSDA                                                    HARRIS
C      INTEGER*4 MSYSDA                                                  IBM-PC
C
      REAL DYCPU1,DYCPU2,FCPU1,FCPU2
C
      COMMON /IO/ NFL,IBF,IEF,LNF,IDT,IDP,ITBL,KDI,KDO,KUTR,KDNME,KDDES,
     .KDTR,MSYSDA,ISYSMN,LBATCH,LERROR,LSTOP,LEOF,LECHO,LNEWBA,LBLANK,
     .LCOMND,LDTK14,LDCALC,LVERIF,LPRMPT,LEXIT,LFINIS,LNEXT,LALL,LCCPL,
     .LHPPL,LVERPL,LZETPL,NPROGN,LOPENP,DAYMON,DYCPU1,DYCPU2,FCPU1,FCPU2
C     ------------------------------------------------------------------
C
C                                                              param
C     ------------------------------------------------------------------
      PARAMETER (KU_PARAM_NAME=15, KU_LOC_NAME=13)
      CHARACTER ID_PARAM_NAME(KU_PARAM_NAME)*15,
     . ID_LOC_NAME(KU_LOC_NAME)*6, ID_RIV_NAME(KU_LOC_NAME)*20,
     . ID_LOC_NAME_FLOW(KU_LOC_NAME)*20,
     . ID_PARAM_PNPRT(KU_PARAM_NAME)*20
      COMMON /PARAM/ ID_PARAM_NAME, ID_LOC_NAME, ID_RIV_NAME,
     . ID_LOC_NAME_FLOW, ID_PARAM_PNPRT
```

Appendix I

MENUPRM and Associated Programs

MENUPRM and Associated Programs

Table of Contents

List of MENUPRM Menus

Purpose of MENUPRM

MENUPRM assists the user in:

- Assigning a subdirectory to each study.

- Selecting a study and moving to the associated subdirectory.

- Selecting the desired computer program.

- Selecting files needed by the selected program. Includes ASCII input-output files, ASCII macro & function files, and binary files such as HEC-DSS data files.

- Listing existing, appropriate files for the selected file in the menu. (F5)

- Editing files using the COED editor. For the HEC-PRM program, the following options are invoked: (1) Full Screen Editing, (2) Help Program feature. (Alt-E)

- Editing files using the alternative editor "Blackbeard" from Lahey Computer Systems. (Alt-B)

- Deleting the highlighted file. (Alt-D)

- Executing the selected program. MENUPRM generates the appropriate command line including any file assignments. (Alt-X)

- Viewing the output from the program. The "LIST.COM" program is used to list ASCII files. It allows convenient viewing of files which have a character width greater than 80. (Alt-L)

- Printing files to the printer. The DOS command "PRINT" is used perserving character column 1. (Alt-P)

- Automatically post-processing results from HEC-PRM using the programs PRMPOST and MATHPK.

Brief Description of MENUPRM Screens

Assigning a subdirectory to each study.

MENUPRM allows you to assign a subdirectory to a study name. After the banner page, select "create new study" to define a new study. There is no online directory tree information --- you must know the subdirectory into which your data will be entered. If the subdirectory does not exist, it will be created.

Selecting a study and moving to the associated subdirectory.

Each time the MENUPRM is started, you may select an existing study. MENUPRM will change to the selected study and physically move you to the associated subdirectory.

Selecting the desired computer program.

After the "select study" list, MENUPRM displays a list of computer programs which are accessable from the menu. The user may select a program by either: (1) entering the number followed by pressing the "Enter" key, or (2) highlighting the program using the cursor, home, or end keys and pressing the "Enter" key.

Selecting files needed by the selected program.

The files which the user defines through MENUPRM include ASCII input-output files, ASCII macro & function files, and binary files such as HEC-DSS data files.

For each program, MENUPRM lists appropriate files needed by the program for input and/or output. The user may enter names for any of the files or all or let them revert to defaults. Some files should be defined as the default is not a good or practical solution. For example, the ASCII HEC-PRM input file should be defined as a file into which fixed-formatted input is entered. The default is the keyboard which not a practical source of HEC-PRM input information.

The files are defined by moving the highlighted cursor to the desired file and entering the filenames.

Listing existing, appropriate files for the selected file in the menu.

Existing files may be displayed in a box on the screen and the desired file may be selected by cursoring to that file and pressing the "enter" key. The list is obtained by highlighting the desired type of file in the "define data files" menu and then pressing the "F5" function key. All files have an associated filename extension. For example, the HEC-PRM ASCII input data files all have the extension ".PRI" and the output data files the extension ".PRO". When the user presses the F5 key for HEC-PRM input, MENUPRM will list all files that have the extension ".PRI".

Editing files using the COED editor.

When editing files using the COED text editor with the HEC-PRM program, the following options are automatically invoked by MENUPRM: (1) Full Screen Editing, (2) Help Program feature. To edit an ASCII file with COED, highlight the file in the "define data files" menu and press the keys **"Alt-E"**. If the file is some other file besides the HEC-PRM ASCII input data file, MENUPRM invokes COED in full screen but without the help program feature.

A new HEC-PRM input data file is created by defing the file, editing the file, and entering records in the proper order. A similar existing file may be used as a starting point for the new file. To use it, it may be either: (1) copied to the new file when you are at the DOS prompt or (2) Retrieved into the new file using the COED command "Get". To "GET" the existing file, edit the new file ("alt-e"), press the line edit mode key F10, enter the command "GET filename" where "filename" is the name of the existing file.

To use the "help Program" feature, you may move the cursor to any record in the HEC-PRM ASCII input data file and press **"Alt-F1"**. If the cursor is positioned on the first two columns of input, COED gives more general information about that record. If it is positioned in some data field, information specific to that data field is displayed.

To get online information about the COED editor, press the **F1** key.

Editing files using the alternative editor "Blackbeard" from Lahey Computer Systems.

COED works very poorly in full screen mode if data files contain more than 80 characters of information. Moreover, if you start with a file which contains less than 81 characters but add information beyond 80 characters, the additional characters will be lost. While this is not a problem for HEC-PRM ASCII input, it can be a problem for other files such as the PREAD macro files. While the Blackbeard editor probably will not be included with HEC-PRM distribution, some means of invoking an alternative editor to COED will be provided. To invoke the Blackbeard editor, the filename is highlighted in the "define data files" menu and the keys **"Alt_b"** are pressed.

Deleting the highlighted file.

A file may be deleted by highlighting it in the "define data files" menu. To delete a file, press the **"Alt-d"** keys. MENUPRM will ask you to confirm that you wish to delete the selected file.

Executing the selected program.

MENUPRM generates the appropriate command line including any file assignments.

The selected program may be executed from either the "select program" menu or the "define data files" menu. To execute the selected program, press the "Alt-x" keys. On the "select program" menu, the cursor must highlight the desired program. On the "define data files" menu, the cursor may be positioned on any filename or on the "default extension" block.

Viewing the output from the program.

The "LIST.COM" program is used to list ASCII files. It allows convenient viewing of files which have a character width greater than 80.

To view output from "batch" program such as HEC-PRM, highlight the output results file name and press the keys "Alt-L". The MENUPRM will invoke the "LIST.COM" program so that you can view the output from the program. Cursor keys control up, down, left, and right movement. PgUp and PgDn move in paging fashion. Other controls are available. Limited online information is available by pressing the F1 key.

If you try to "list" an HEC-DSS data file, MENUPRM will instead try to list the associated catalog file. For example, if you try to list the file "MRD.DSS", MENUPRM will instead try to list the file "MRD.DSC".

Printing files to the printer.

The DOS command "PRINT" is used perserving character column 1. Any file listed in the "define data files" menu may be sent to your printer by pressing the "Alt-P" keys. You should use caution in doing this as some files are quite large and it is not practical to print them.

Automatically post-processing results from HEC-PRM using the programs PRMPOST and MATHPK.

When you execute HEC-PRM, MENUPRM will ask you "Do you want to automatically post-process HEC-PRM results". If you answer yes, MENUPRM generates commands in the file "runfile.bat" to execute HECPRM.EXE, PRMPOST.EXE, and MATHPK.EXE. Program PRMPOST.EXE generates output similar to MRD'S long range planning model program. Program MATHPK calculates parameters and tabulates results to a tabulation file in a vertical format. In addition to adding the command lines for post-processing, MENUPRM generates input data for these programs that corresponds to the time window and alternative which is analyzed by HEC-PRM.

PRMPOST has pathname parts hardwired within the program --- modifications require mondifications to the FORTRAN source code.

MATHPK utilizes extensive PREAD macros to define pathname parts and perform computations. The user has access to these macros and may make changes as appropriate.

Programs Associated With HEC-PRM

Currently, 12 programs can be accessed by MENUPRM. They include:

HEC-PRM
PRMPOST
DSS-DSSUTL
DSS-DSPLAY
DSS-MATHPK
DSS-DSSPD
DSS-DSSTS
RDMATF
RDATA0
DSS-DSSTXT
ABSL01 and
V1.

Each program has specific roles to play in the overall computational picture. The following is a brief description of each program's role and it's associated menu.

Figure 30 HEC-PRM Data Relationships

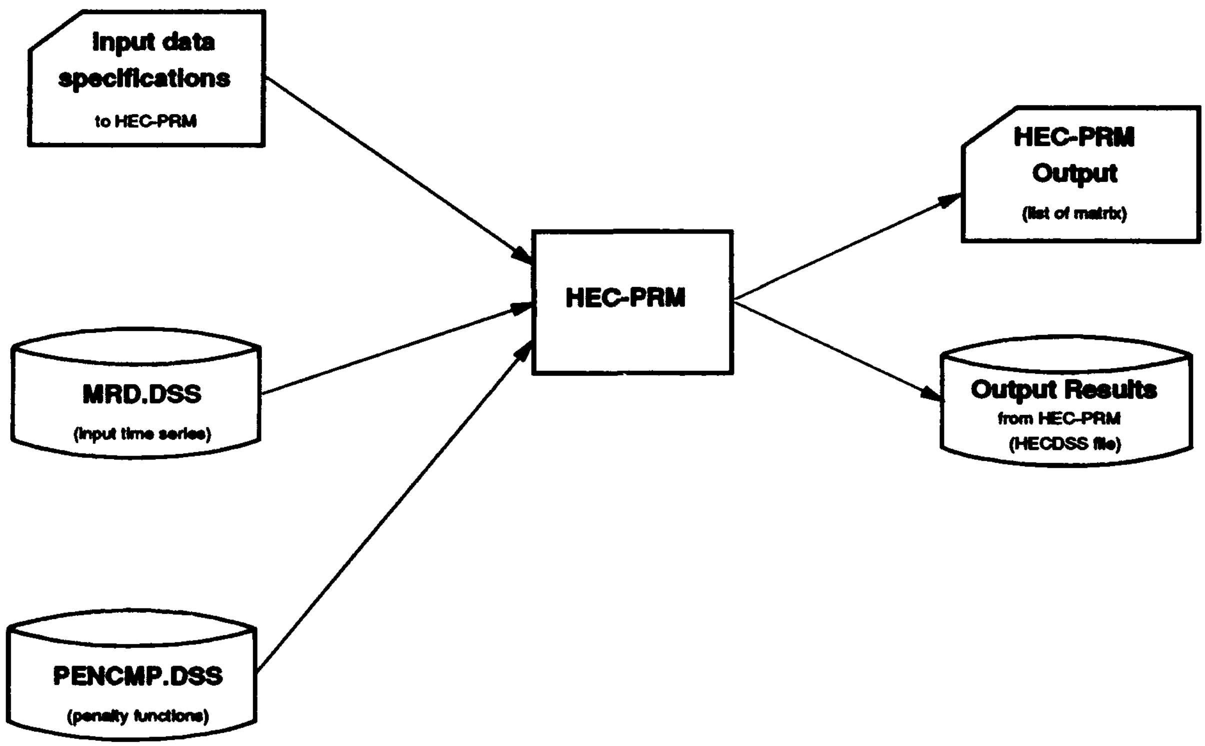


Figure 31 HEC-PRM Input Data Development

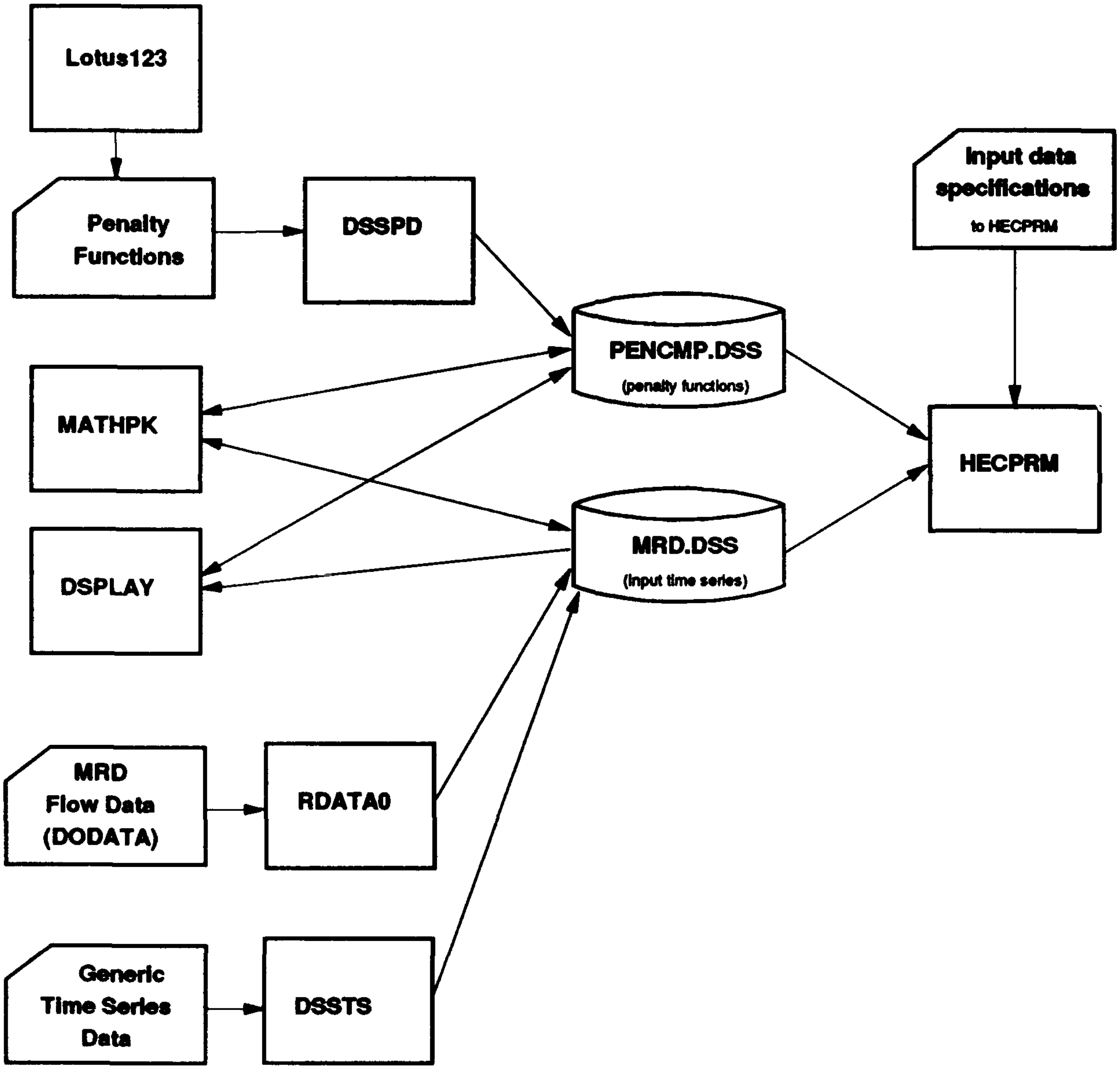


Figure 32 HEC-PRM Post-Processing

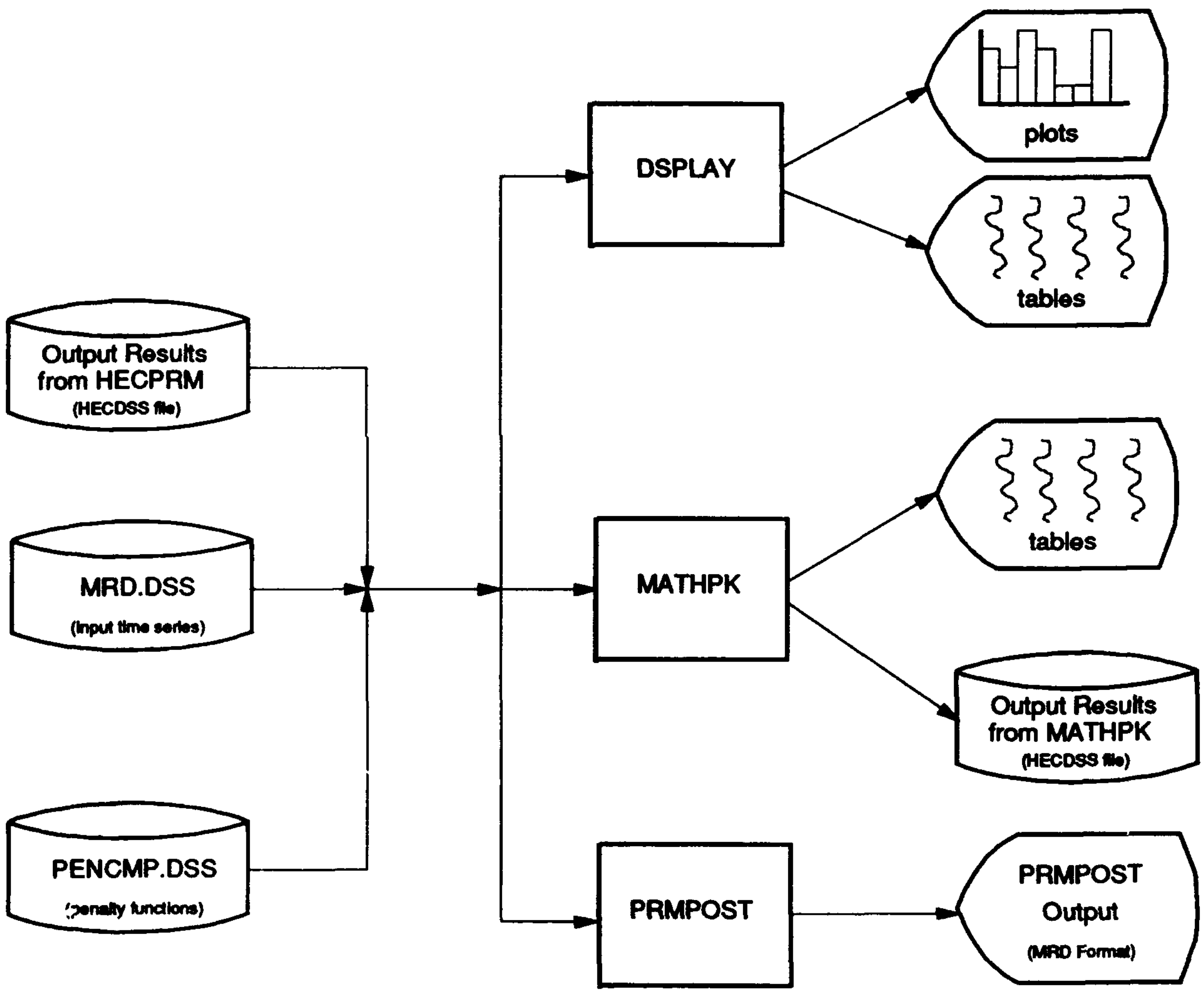
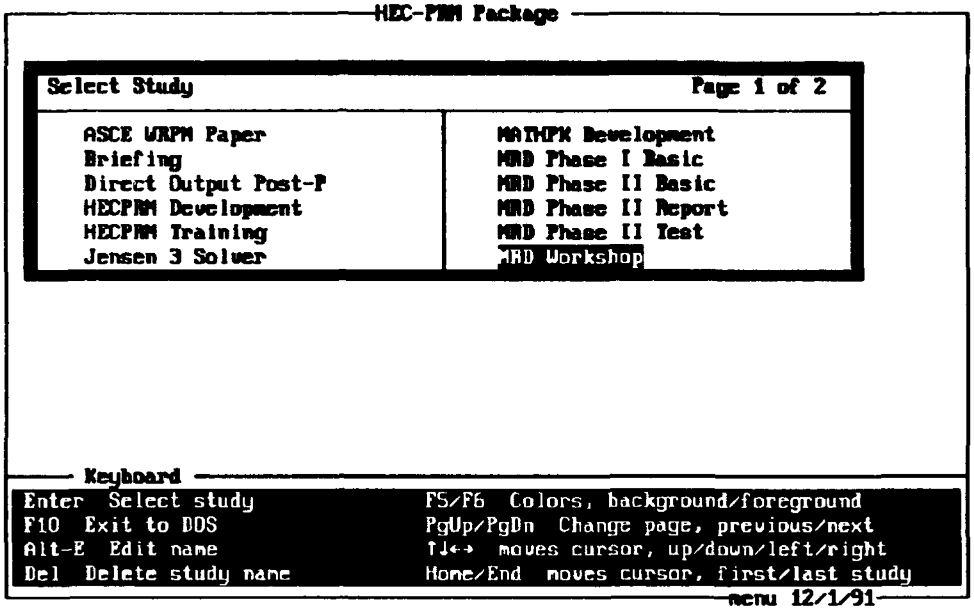


Figure 33 Banner Page

```
---------------------------- HEC-PRM Package ----------------------------



                  Prescriptive Reservoir Model Package


                     Hydrologic Engineering Center
                     U.S. Army Corps of Engineers
                          609 Second Street
                     Davis, California 95616-4687
                            (916) 756-1104

------------------------------------------------------------- menu 12/1/91 ---
```

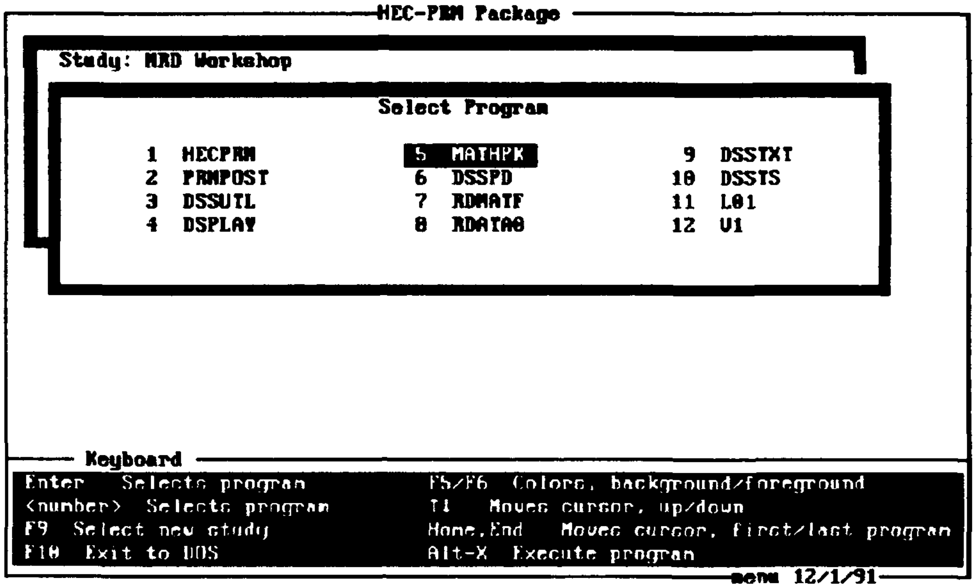
The banner page is displayed every time MENUPRM is executed. The banner will go away after about 10 seconds or if the user presses any key.

Figure 34 Study List Menu



The "select study" menu is displayed after the banner menu. From here, the user may select an existing study or by highlighting the study "select new study", the user may define a new study. For each study, there is an associated subdirectory to which MENUPRM physically places the user upon selection. You should keep all files associated with one study in one subdirectory and you should keep only one study in a single subdirectory.

Figure 35 List of Programs Menu



Following the "select study" menu, MENUPRM displays the "select program" menu. The user may execute any of the programs from this menu if all necessary files for that program have already been defined. However, if any necessary file has not been defined, the user must select the program by either: (1) entering the program's number and pressing the "enter" key, or (2) highlighting the desired program and pressing the "enter" key. MENUPRM the proceeds to the "define data files" menu for the selected program. To execute a program from the "select program" menu, highlight the desired program and press the "Alt-X" keys.

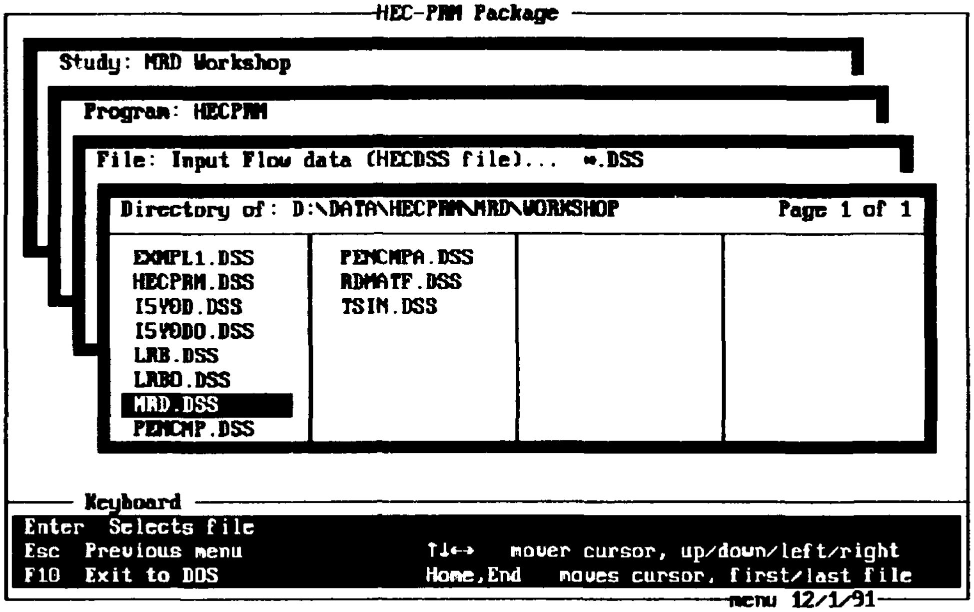
Figure 36 HECPRM Data Files

```
-------------------------------- HEC-PRM Package ------------------------------
  Study: MRD Workshop
    Program: HECPRM
                          Define data files
      Input Data...15Y0D.PRI
      Output Information...15Y0D.PRO
      Results from HECPRM (HECDSS file)...15Y0D.DSS
      Input Flow data (HECDSS file)...MRD.DSS
      Input Penalty Functions (HECDSS file)...PENCMP.DSS

      Default extensions...Yes

------ Keyboard ----------------------------------------------------------------
 F5  File directory                    F2  reset filenames to initial definition
 Keyboard  Edit filename               ↑↓   moves cursor, up/down
 F10  Exit to DOS                      Home,End   moves cursor, first/last option
 Alt menu:  (D)elete   (E)dit   (L)ist   (P)rint   e(X)ecute
---------------------------------------------------------------- menu 12/1/91 --
```

The HEC-PRM model requires the user to define 5 files --- they all must be defined. The Input Data file contains data in an ASCII format. The user normally invokes the COED editor to enter information in this file. The information includes a list of nodes and links, the time window, time series pathnames, and penalty function pathnames. When this file is defined, the Output Information file and the Results from HEC-PRM file are defined by automatically appending the default file extensions to the input data file name.

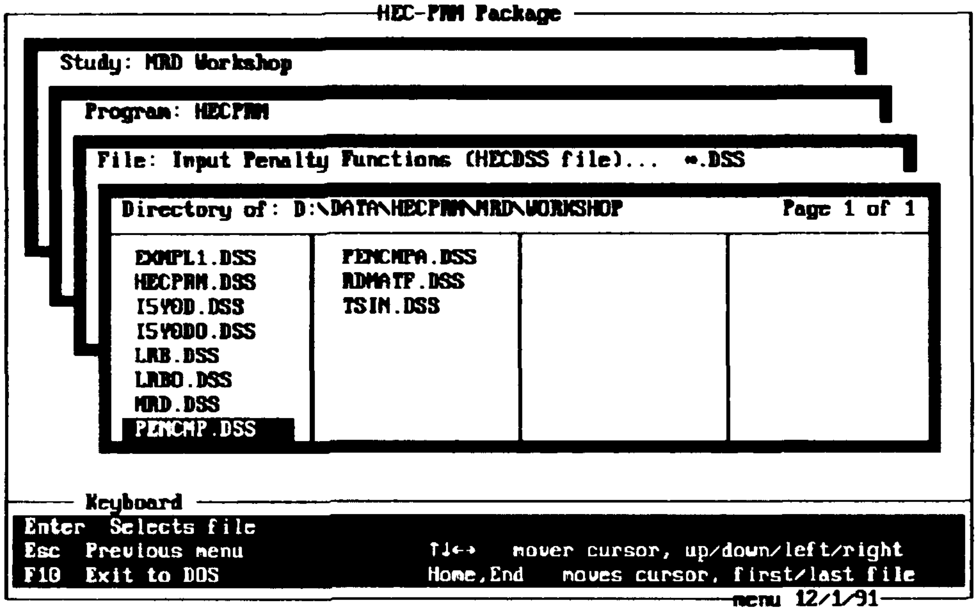
The input flow data is stored in the HEC-DSS data file "MRD.DSS" and the penalty functions are stored int he HEC-DSS data file "PENCMP.DSS". These files can be easily be defined by getting a list of all DSS data files by pressing the "F5" function key.

Figure 37 List of Pertinent Files



The input flow data is stored in the HEC-DSS data file "MRD.DSS" and the penalty functions are stored int he HEC-DSS data file "PENCMP.DSS". These files can be easily be defined by getting a list of all DSS data files by pressing the "F5" function key.

Figure 38 List of Pertinent DSS Data Files



The input flow data is stored in the HEC-DSS data file "MRD.DSS" and the penalty functions are stored int he HEC-DSS data file "PENCMP.DSS". These files can be easily be defined by getting a list of all DSS data files by pressing the "F5" function key.

Figure 39 PRMPOST Data Files

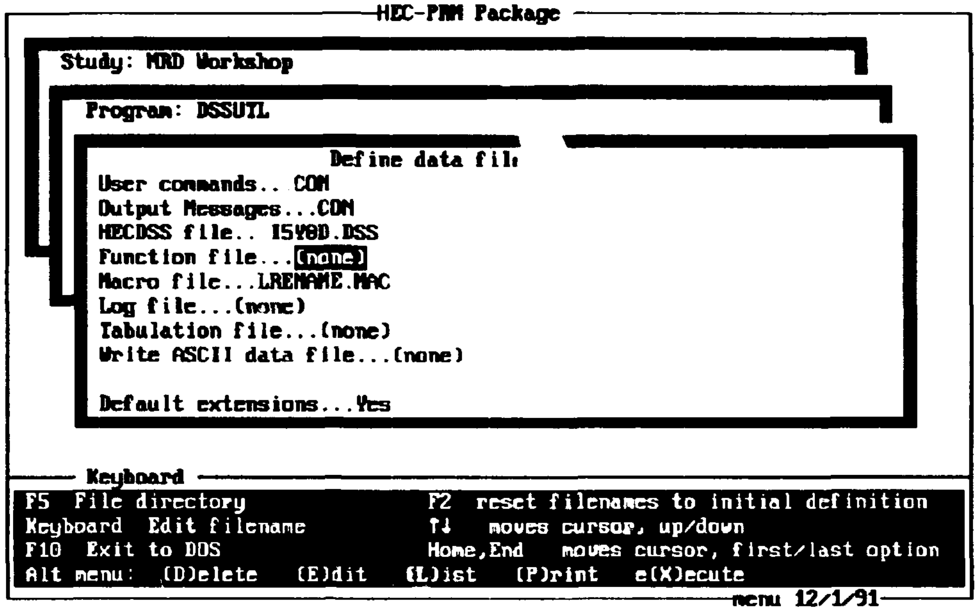
```
HEC-PRM Package
Study: MRD Workshop
Program: PRMPOST
                    Define data files
Input Data...I5Y0D.P01
Output Information...I5Y0D.P00
Tabular Output...I5Y0D.TAB
Results from HECPRM (HECDSS file)...I5Y0D.DSS
Input Flow data (HECDSS file)...MRD.DSS
Input Penalty Functions (HECDSS file)...PENCMP.DSS

Default extensions...Yes

Keyboard
F5  File directory                 F2  reset filenames to initial definition
Keyboard  Edit filename            ↑↓  moves cursor, up/down
F10  Exit to DOS                   Home,End   moves cursor, first/last option
Alt menu:  (D)elete   (E)dit   (L)ist   (P)rint   e(X)ecute
                                                            menu 12/1/91
```

The PRMPOST program requires the user to define 6 data files. However, all of the files are automatically defined when the user specifies the 5 required files for the HEC-PRM program. The input flow data file and the input penalty function files are the same, and the other files are defined by appending the appropriate default extensions to the input data file from HEC-PRM.

Example input to post-process HEC-PRM results:

```
ALT     5Y0D
TIME      MAR1965 FEB1970
GO
```

Figure 40 DSSUTL Data Files



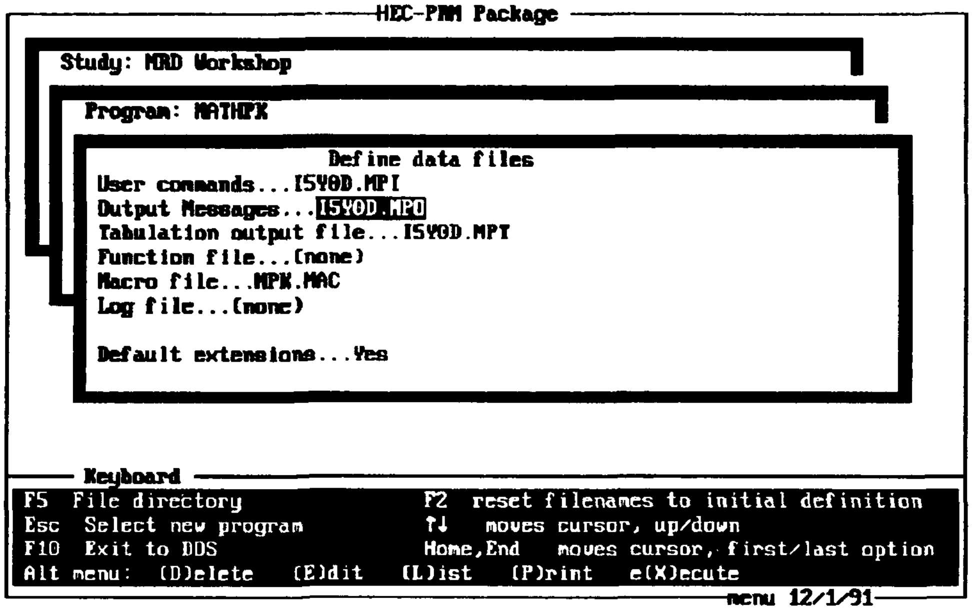
The primary DSSUTL file that the user must define is the DSS data file although if it is left blank, DSSUTL will prompt the user for a filename. The other files are optional and some of them may be defined by DSSUTL input.

Figure 41 DSPLAY Data Files

```
HEC-PRM Package
Study: MRD Workshop
Program: DSPLAY
                    Define data files
User commands...CON
Output Messages...CON
HECDSS file...I5Y8D.DSS
Tabulation output file...(none)
Screen file...(none)
Function file...(none)
Macro file...DSP.MAC
Menu file...(none)
Log file...(none)

Default extensions...Yes
Keyboard
F5  File directory              F2  reset filenames to initial definition
Keyboard  Edit filename         ↑↓  moves cursor, up/down
F10  Exit to DOS                Home,End  moves cursor, first/last option
Alt menu:  (D)elete  (E)dit  (L)ist  (P)rint  e(X)ecute
                                                          menu 12/1/91
```

The primary DSPLAY file that the user must define is the DSS data file although if it is left blank, DSPLAY will prompt the user for a filename. The other files are optional and some of them may be defined by DSPLAY input.

Figure 42 MATHPK Data Files



MATHPK requires none of the files be defined before execution. However, for postprocessing of HEC-PRM results, MENUPRM automatically defines an input data and output information file and even modifies the input data file to provide appropriate information. Also critical to post-processing is the macro file "MPK.MAC". It must be defined by the user before executing the MATHPK program.

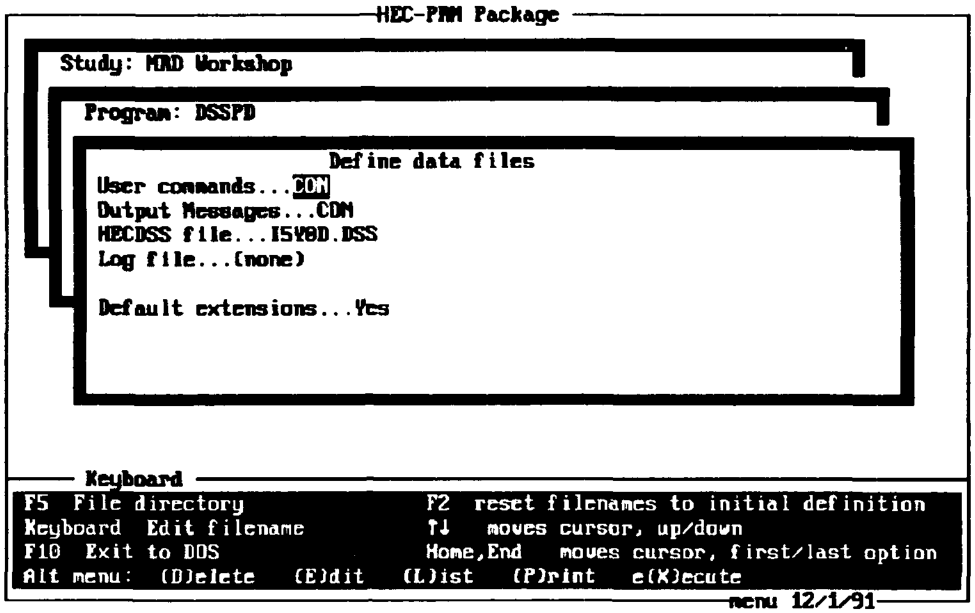
Example input to post-process HEC-PRM results:

```
!RUN DO_PRM 'I5Y0D' 'PENCMP' 'I5Y0DO' 'MRD' '5Y0D'
31MAR1965     28FEB1970
```

Example input to compute adjusted local inflows:

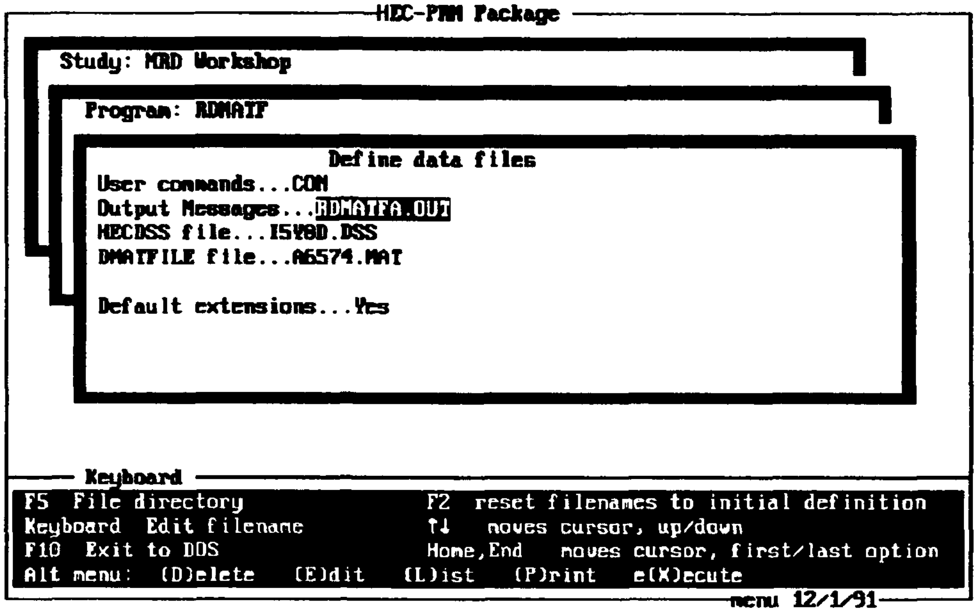
```
OP MRD 1
OP HECPRM 2
TI 31MAR1898 28FEB1990 2400 2400
!RUN FLOW_LOC 1991
FI
```

Figure 43 DSSPD Data Files



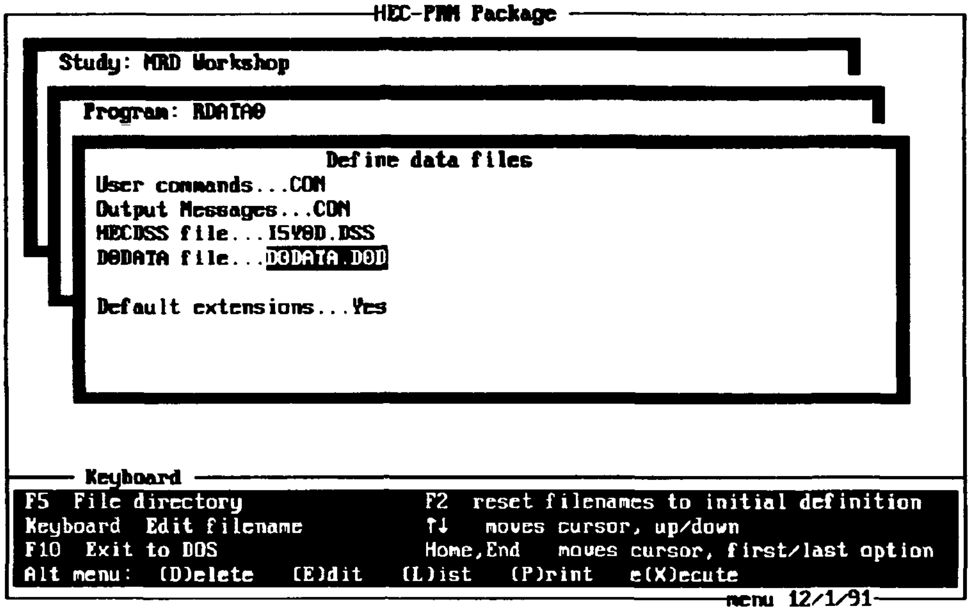
The DSSPD program is used to enter penalty functions, hydropower energy coefficients, and elevation-area-capacity curves into DSS data files. For MRD, this data has been entered in the file "PENCMP.DSS". It is also used to enter the monthly varying depletions into the MRD.DSS data file. When penalty functions are developed in Lotus123, a .prn file is created in the format required by the DSSPD program. In that case, the user would define the input data file to be the .prn file created by Lotus123.

Figure 44 RDMATF Data Files



RDMATF program reads the "DMATFILE" which is ASCII output from MRD's long range planning model "ABSL01". MENUPRM adds a little twist by requiring the "DMATFILE"s have the extension ".mat". This allows the user to make multiple runs of V1.exe creating multiple "DMATFILE"s and then running RDMATF, once for each file. The extension also allows the user to pick "DMATFILE"s from a list generated by a DOS masked directory command ("dir *.mat").

Figure 45 RDATA0 Data Files



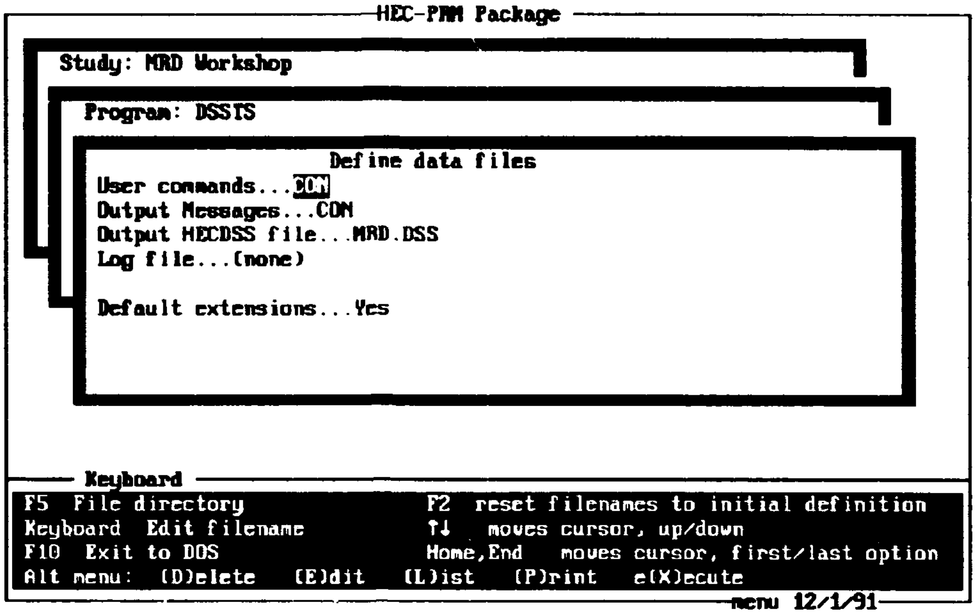
The RDATA0 program reads MRD's D0DATA file and stores evaporation rates, inflow, and depletions in the DSS data file "MRD.DSS". The user must define the DSS data file (which is already defined if the user defined data files for the HEC-PRM program) and the "D0DATA" filename. MENUPRM adds a little twist because it requires the "D0DATA" file to have the extension ".D0D" (with a zero). This allows the user to store on disk several files each having different levels of depletions. The only input required of the user is to enter the year of depletions, everything else is read from the "D0DATA" file.

Figure 46 DSSTXT Data Files

```
HEC-PRM Package
Study: MRD Workshop
Program: DSSTXT
                    Define data files
User commands...CON
Output Messages...CON
HECDSS file...I5Y0D.DSS
Direction...STORE
Text file...(none)
Screen file...(none)
Function file...(none)
Macro file...(none)
Log file...(none)

Default extensions...Yes
Keyboard
F5  File directory              F2  reset filenames to initial definition
Keyboard  Edit filename         ↑↓  moves cursor, up/down
F10  Exit to DOS                Home,End  moves cursor, first/last option
Alt menu:  (D)elete   (E)dit   (L)ist   (P)rint   e(X)ecute
                                                        menu 12/1/91
```

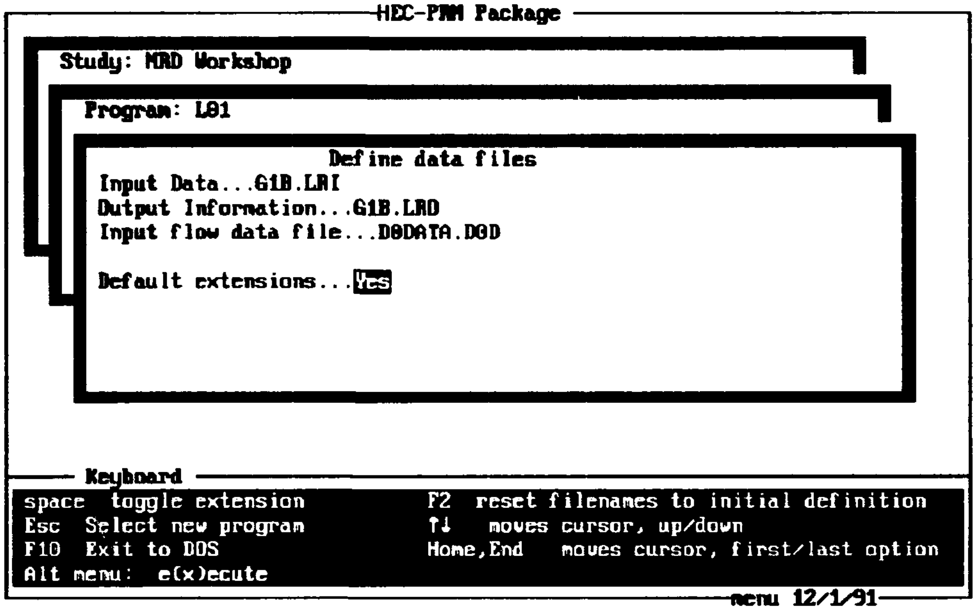
The DSSTXT program is a generalized program that allows the user to store text information in DSS data files. Currently, the text capability of DSS is not being used in any of the HEC-PRM specific programs.

Figure 47 DSSTS Data Files



The DSSTS program facilitates the entry of any regular interval time series data to a DSS data file. For the MRD study, it was not used because all of the necessary time series information was entered using the RDATA0 program to automatically store existing data in the DSS data file.

Figure 48 MRD's Long Range Planning Model (L01) Data Files



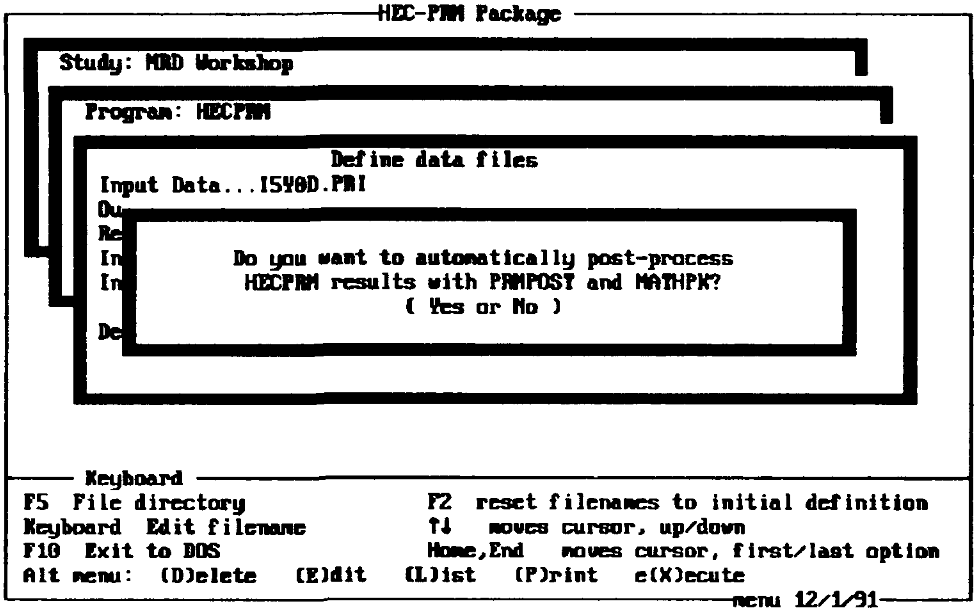
MRD's long range study program "ABSL01" normally reads from the G1 and D0DATA files, and writes to the LRS1OUT files. By calling the file "L01.BAT", the input files and the output file can be defined using the MENUPRM. The example shows its application to define files for the time period 1965-1974 (file G1B.LRI).

Figure 49 MRD's Post-Processor V1.EXE Data Files

```
HEC-PRM Package
Study: MRD Workshop
Program: V1
                    Define data files
User commands...CON
Input Information from L01 (LRS1OUT)...G1B.LRO
Output DMATFILE for graphs and DSS...A6574.MAT
Location / Parameter selection list...DEPL.GRF

Default extensions...Yes

Keyboard
F5  File directory                  F2  reset filenames to initial definition
Keyboard  Edit filename             ↑↓   moves cursor, up/down
F10  Exit to DOS                    Home,End   moves cursor, first/last option
Alt menu:  (D)elete   (E)dit   (L)ist   (P)rint   e(X)ecute
                                                         menu 12/1/91
```

MRD's program V1.EXE normally reads from the files LASTGR.GRF and LRS1OUT which is created by the ABSL01 program and writes information to the "DMATFILE". MENUPRM allows the user to define these files with the extensions ".GRF", ".LRO", and ".MAT" respectively. MENUPRM creates DOS commands to copy the x.grf file into LASTGR.GRF, the x.lro file into LRS1OUT before executing V1. After V1 is through, the LASTGR.GRF file is copied back to x.grf and the DMATFILE is copied into the x.mat file. The use of the file extension is compatible with the extensions used for "L01.BAT".

Figure 50 Example Prompt to Post-Process HEC-PRM Results



This is the prompt MENUPRM displays before executing HEC-PRM. To post-process HEC-PRM results, the most desirable option is to enter "yes" (y) to this prompt. MENUPRM automatically generates input for the PRMPOST and MATHPK programs and will automatically execute them after HEC-PRM is complete.